MorphoDig User's guide
MorphoDig v1.0

Renaud LEBRUN

Institut des Sciences de l'Evolution, University of Montpellier, France

July 18, 2018

Contents

Introduction

Contents

MorphoDig is based on the design concepts of the software FoRM-IT (Fossil Reconstruction and Morphometry Interactive Toolkit; [Zollikofer and Ponce de León, 1995, 2005]). MorphoDig[?] was developed as a help to the scientific journal MorphoMuseuM (M3), in order to help scientists to produce enriched surface models. The source code is hosted on Github.

Origin of the project

Over the last two decades, even though 3D data acquisition and computer-assisted techniques have grown increasingly popular among biologists, paleontologists and paleoanthropologists, so far, no standard biology-oriented 3D mesh manipulation software has emerged; most of the time, researchers either use commercial software which are not primarily designed for biologists, or develop their own in-house software solutions. MorphoDig builds upon the design concepts of the software FoRM-IT (Fossil Reconstruction and Morphometry Interactive Toolkit; [Zollikofer and Ponce de León, 1995, 2005]), and is developed to meet the need to ease the production and the diffusion of 3D models of biological organisms. MorphoDig provides a set of tools for editing, positioning, deforming, labeling, tagging sets of 3D surfaces. As such, MorphoDig can be used to produce enriched models which can in turn be submitted to M3.

Main features

Features include:

- Retro-deformation for virtual restoration of fossils/deformed specimens;

- Point and curve primitives for placing the exact type of landmark points you're interested in
- Easy to use 3D interface for positioning and manipulating sets of surfaces and landmark primitives
- Mesh tagging, labeling and coloring (to allow for the creation of anatomy atlases)
- Mesh scalar computation and coloring (based upon curvature/thickness etc...)

MorphoDig allows to import and export 3D meshes in standard formats such as STL, PLY and VTK polydata, and as such, in can be used in conjunction with a variety of other 3D mesh editors such as MeshLab (http://meshlab.sourceforge.net/) or Blender (https://www.blender.org/)

Implementation

MorphoDig is entirely written in C++, and uses the visualization library VTK [Avila et al., 2001]. The GUI has been designed with QT (https://www.qt.io/). MorphoDig is open-source and cross platform, and we are looking forward to welcoming new developers in the future in order to implement new functionalities.

Chapter 1

Licence

Contents

1.1 MorphoDig

1.2 VTK

Chapter 2

F.A.Q.

Contents

2.1 How should I cite MorphoDig in scientific publications ?

You may cite MorphoDig with the following reference :
Lebrun, R. MorphoDig, an open-source 3D freeware dedicated to biology. IPC5, Paris, France; 07/2018.

2.2 Is MorphoDig a geometric morphometrics software ?

No. However, you can digitize 3D landmarks on complex 3D surfaces using MorphoDig, which you can use in other software.

2.3 Can I produce/extract 3D meshes out of CT/MRI data using MorphoDig ?

No. To extract 3D meshes CT/MRI data sets, you have to use another software. However, you can edit 3D Meshes in various ways using MorphoDig.

2.4 Is there a CTRL-Z functionality ?

Yes, definitely. Most actions can be undone and redone.

Chapter 3

Interaction modes

Contents

3.1 Object selection... and what color is currently displayed?

Selected objects (3D surfaces and landmarks) are always “grey". All currently opened objects can be selected by pressing CTRL+A. All currently opened objects can be unselected by pressing CTRL+D. Objects can also be selected/unselected using the right mouse button, depending on the currently active interaction mode (see below). A given unselected surface can be colored:

- using a uniform “solid" color. Scalar display mode must be deactivated or active array combo box set to “Solid color" (Solid color)
- according to the tag values (= integers) associated to the vertices. Array display mode button must be pressed ( ), and a Tag array must be selected (ex:ConnectivityTags).
- according to the scalar values (= numbers) associated at the vertices. Array display mode button must be pressed ( ), and a Scalar array must be selected (ex:Thickness)
- according to RGB values associated to the vertices. Array display mode button must be pressed ( ), and a RGB array must be selected (ex:RGB)

3.2 Interaction modes

3.2.1 Camera mode

"Camera mode" is the default interaction mode, and is active on startup. When active, left and middle mouse button drags result in camera rotation/translation, respectively. Right mouse button drag results in object selection/unselection. Pressing "ESC" switches between object mode and camera mode.

3.2.2 Object mode

When active, left and middle mouse button drags result in object rotation/translation, respectively. Right mouse button drag results in object selection/unselection. Pressing "ESC" switches between object mode and camera mode.

3.2.3 Landmark mode

When active, only landmarks can be selected/unselected via right mouse button drag. This mode is useful when editing/placing landmarks. Left and middle mouse button drags result in camera rotation/translation, respectively.

3.3 Landmark setting modes

Landmarks can be set on surfaces by pressing "L" + left mouse click. Four series of landmarks can be set with MorphoDig: "normal" landmarks, "target" landmarks, "curve node" landmarks and "curve handle" landmarks. Additionally a fourth special landmark series ("flag" landmarks) can be used to label surface structures.

3.3.1 Normal landmark mode

Press "" to activate this mode (this mode is active by default)

3.3.2 Target landmark mode

Press "" to activate this mode

3.3.3 Curve node mode

Press "" to activate this mode

3.3.4 Curve handle mode

Press “" to activate this mode

3.3.5 Flag landmark mode

Press “ " to activate this mode

Chapter 4

Keyboard and mouse controls

Contents

4.1 Keyboard and mouse controls

Ctrl+A	Selects all objects
Ctrl+D	Unselects all objects
L+ left click	Creates a landmark (either “normal", “target", “curve node", “curve handle" or “flag" landmark).
L + right click	If a single landmark is selected, its position is moved. Nothing happens if no landmark is selected or if more than one landmark are selected
Left mouse button drag	Camera mode : camera rotation. Object mode : object rotation. Landmark mode : camera rotation.
Ctrl + left mouse button drag	Camera mode : object rotation. Object mode : camera rotation. Landmark mode : object rotation.
Right mouse button drag	Draws a yellow rectangle. Once right button is released, all objects (surfaces and landmarks) falling inside the rectangle get selected/unselected, depending on their initial selection status
Middle mouse button roll (roll wheel)	Zoom / unzoom
Middle mouse button drag	Camera mode : camera translation Object mode : object translation Landmark mode : camera translation
Ctrl + middle mouse button drag	Camera mode : object translation Object mode : camera translation Landmark mode : object translation
« Del »	All selected objects are deleted
T + left click	Picks a vertex and tags corresponding surface with active tag value (a tag array must be active, and tag mode must be activated)
T + right click	Picks a vertex and tags corresponding surface with active tag value (a tag array must be active, and tag mode must be activated). The only potentially affected vertices are those having initially the same color as that of the picked vertex.

4.2 Additional controls

Additional controls are available when using “lasso cut" or “lasso tag" (lasso mode should be active):

Left mouse click and drag	Draws the lasso polygon contour
Left mouse release	all the vertices falling within the polygonal region (outside or inside the polygon) are either: - given the color corresponding the active tag (lasso tag) - inserted into a new object (lasso cut inside). - deleted from the new object (lasso cut outside). See lasso cut (section 8.1.5) and lasso tag (section 11.1.5) sections for further information.

Chapter 5

Camera and object GUI main controls

Contents

5.1 Camera controls

5.1.1 Camera rotation center

By default, the camera rotates around the origin of the coordinate system (x=0, y=0, z=0), but by pressing “", the camera will revolve around the center of mass of all opened objects. The latter option is useful when the centre of mass of an object (or of several ones) is far from the origin of the coordinate system. The grid is drawn using different colors depending on the camera rotation centre (see Fig. 5.1).

5.1.2 Orthographic or perspective projection

You can switch between orthographic and perspective projection mode by pressing the "" or "" toggle button. Note that in orthographic projection mode , some display information regarding the grid and pixel size appear on the right bottom corner of the screen : Grid: 1 square=5mm, 100px=37.35mm.

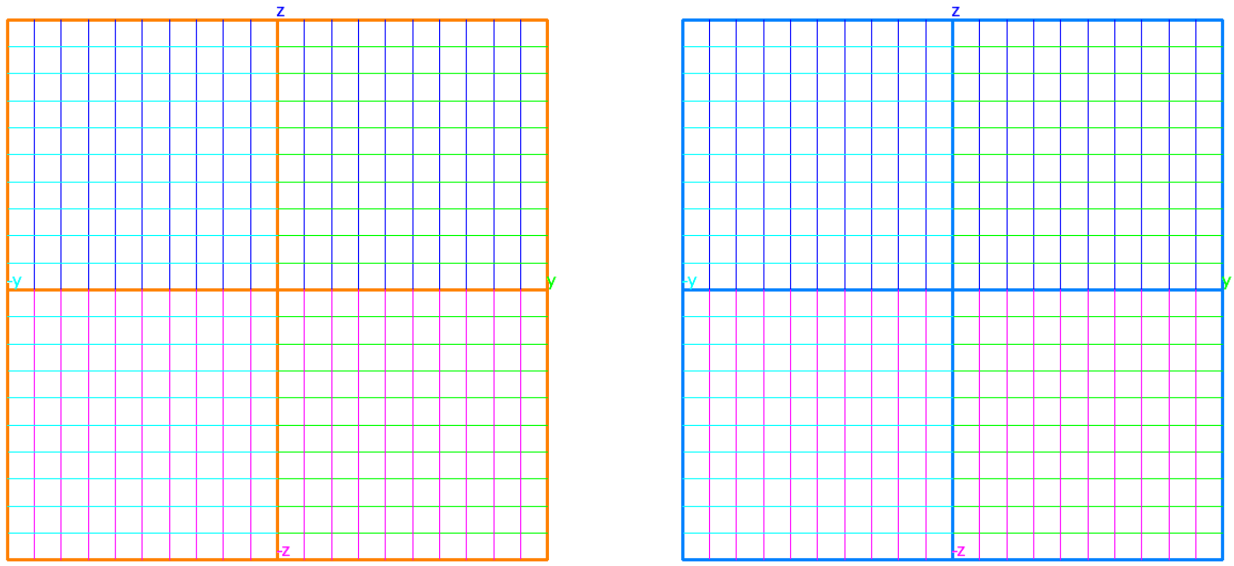


Figure 5.1: Grid display color. Left: when the camera revolves around the origin of the coordinate system (x=0, y=0, z=0), the grid outline is displayed in orange. Right: when the camera revolves around the center of mass of all opened objects, the grid has a blue outline.

5.1.3 Camera orientation

6 camera positions are predefined :

view object from right side

view object from left side

view object from front side (default camera position)

view object from back side

view object from above

view object from below

5.1.4 Camera rotation around “z" viewing axis

To do so, you may use the slider lying around the center of the left panel of the main window.

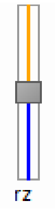


Figure 5.2: Camera “z" rotation slider

5.1.5 Clipping plane

In some cases, you may need to displace the viewing clipping plane. To do so, use the slider lying centrally in the left panel of the main window.

The buttons and which lie just underneath the clipping plane slider (and also in the camera option window) also permit to adjust / readjust the position of the clipping plane at predefined positions :

- : the clipping plane is placed at z = 0 (all objects having a z coordinate along z viewing axis smaller than 0 are hidden).
- : the clipping plane is replaced at its original value : z= - camera.far / 2. This value permits to view objects having positive and negative coordinates along z viewing axis.

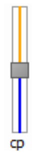


Figure 5.3: Camera clipping plane slider

5.1.6 Zoom

There are three main ways to modify the “zoom" in MorphoDig :

- You may use the zoom slider laying in the lower part of the left panel of the main window.
- You may set manually the display scale (Edit → Edit size unit and grid spacing, then define the display scale: 100 pixels in size unit). This option is only available in orthographic projection mode .
- You may use the middle click mouse roll button (roll the wheel).

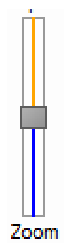


Figure 5.4: Zoom slider

When the option "Adapt field of view depth" is active in the Rendering options window (Viewing opt.→General rendering options → Depth of field of view panel), changing the zoom value will also modify the depth of the field of view (camera.far value) and the position of the clipping plane (camera.tz value). When the option "Keep current field of view depth" is active, changing the zoom will not affect the camera.far and camera.tz values.

5.2 Other display controls

5.2.1 Coordinate system orientation helper

Press to show / hide the coordinate system orientation helper lying on the bottom left corner of the main 3D window. By default, the labels are defined the following way:
+z axis : dorsal side
-z axis : ventral side
+y axis : left side
-y axis : right side
+x axis : proximal side
-x axis : distal side.
You may edit these labels depending on your preferences (for instance, depending on the structure you are working with, you may need to set "+y" to "labial", and "-y" to "lateral"). To edit orientation labels, click on "Edit → Edit orientation labels."

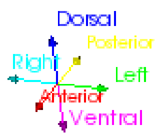


Figure 5.5: Orientation helper

5.2.2 Grid

Press to show / hide the grid. Default grid size is 1 cm / square. Grid size can be edited manually (viewing opt. → Grid size). Switching between the 6 camera predefined positions defined above (,,, , and)will affect the plane in which the grid is drawn.

5.2.3 Lightning

6 lightning orientations are predefined :
light from right viewing side
light from left viewing side
light from front viewing side
light from back viewing side
light from above
light from below

5.3 Object controls controls

As seen earlier, selected objects can be translated and rotated using the mouse left and middle buttons (in landmark and camera selection modes, you also need to maintain “CTRL" button pressed while dragging the mouse to achieve rotation and translation of selected objects). Alternatively, you may also use the following controls to accomplish rotation and translation of selected objects. Rotation is performed around the global center of mass of all selected objects.

5.3.1 Rotation around and translation along “z" viewing axis

These controls are extremely useful, as there is no way to achieve rotation around « z » viewing axis or translation along “z" viewing axis using the mouse.
To do so, use the slider and roller lying in the upper part of the left panel of the main window.

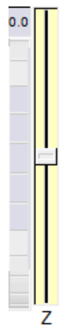


Figure 5.6: Object “z" rotation roller and slider

5.3.2 Rotation around “x" and translation along “y" viewing axes

To do so, use the slider and roller lying in the lower part of the left panel of the main window.

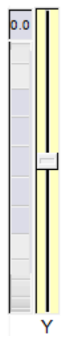


Figure 5.7: Object “x" roller and “y" slider

5.3.3 Rotation around “y" and translation along “x" viewing axes

To do so, use the slider and roller lying in the left part of the bottom panel of the main window.

Figure 5.8: Object “y" roller and “z" slider

Chapter 6

Menu File

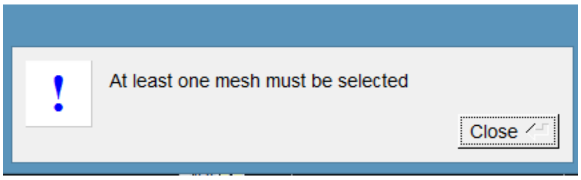
Contents

6.1 Open surface

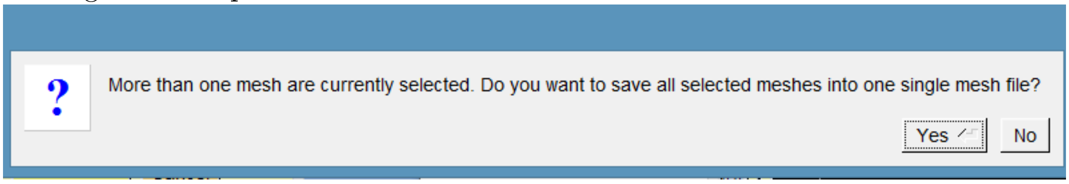
.stl, .ply, .vtk surfaces can be open via this menu. ISE-MeshTools does not manage textures associated with meshes; still, you can open .obj files, but associated textures will not be loaded. When opening a .ply file containing RGB colors (for instance a file painted manually or automatically with "MeshLab") or a .vtk file containing RGB scalars, these colors are placed inside the "RGB" scalars. MeshTools will reinitialize the "RGB" scalars whenever you change the object's color or whenever you activate tag display mode or scalar display mode : if these colors are important to you, you may convert them to TAG values in the menu Tags→Convert RGB colors to tags before changing the object color or display mode/before changing tags or any color scale activated.

6.2 Save surface

Selected surfaces can be saved into files. If no surface is selected, the following message appears:



If more than one mesh are selected, the following message shows up:



6.2.1 Save .ply

Options:

- File type: you can save .ply data in binary (little or big endian) or ASCII formats.
- Position : you can keep object original coordinate system or save the surface in its current position.
- Normales : you can chose whether you wish to save normales.

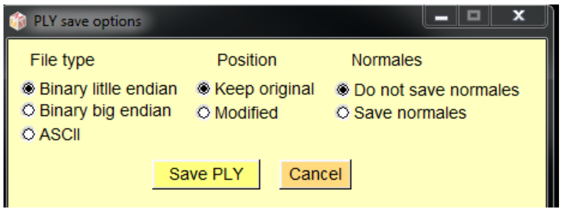


Figure 6.1: PLY save options window

Note that the "RGB" scalar (object rendering color, depending on which rendering mode you are using) will be saved inside the .ply file. This means that Tag / Scalar / Object color can be exported and viewed in other software such as MeshLab.

6.2.2 Save .stl

Options:

- File type: you can save .stl data in binary (little endian) or ASCII formats.
- Position : you can keep object original coordinate system or save the surface in its current position

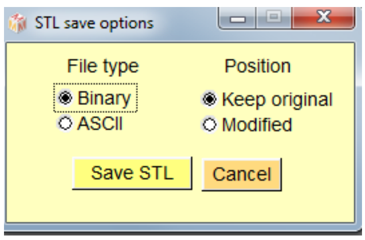


Figure 6.2: STL save options window

6.2.3 Save .vtk

Vtk mesh file format is by far not as widespread as stl or ply format. However, it is extremely useful as it allows to store scalar and tag values at each vertex or at each triangle. Options:

- File type: you can save .vtk data in binary (little endian) or ASCII formats.
- Position : you can keep object original coordinate system or save the surface in its current position

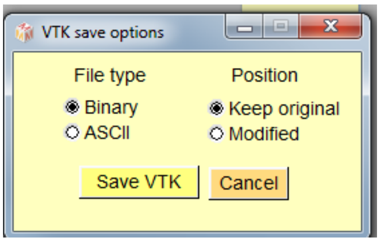


Figure 6.3: VTK save options window

Note that the “RGB" scalar (object rendering color, depending on which rendering mode you are using) will be saved inside the .vtk file.

6.2.4 Save .obj

ISE-MeshTools does not manage textures associated with meshes. Still, you can save meshes in .obj format, but textures will not be saved. Options:

- Position : you can keep object original coordinate system or save the surface in its current position

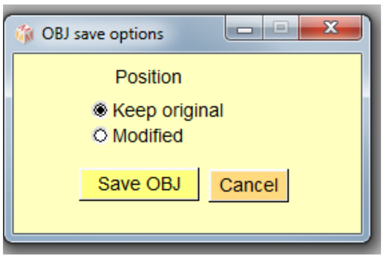


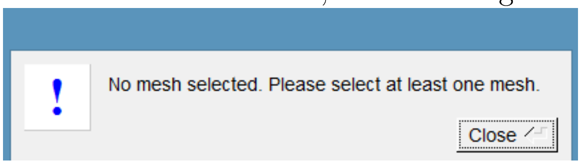
Figure 6.4: OBJ save options window

6.3 Position

In ISE-MeshTools, mesh position consists in two 4*4 square matrices: one matrix is used as the aspect matrix (by default the identity matrix), and the other one as the position matrix. These matrices can be opened and saved in “.pos" format (see Fig. 6.5).

6.3.1 Load position

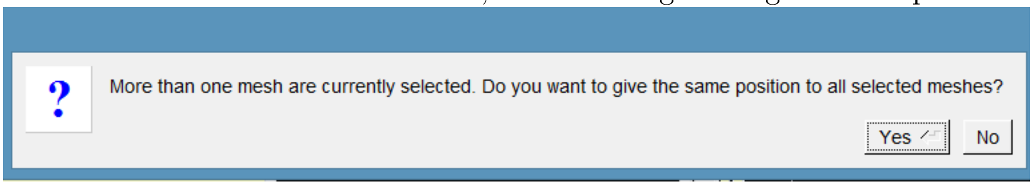
If no surface is selected, the following message appears:



```
1.000000 0.000000 0.000000 0.000000
0.000000 1.000000 0.000000 0.000000
0.000000 0.000000 1.000000 0.000000
0.000000 0.000000 0.000000 1.000000
-0.035449 -0.919797 -0.390790 0.000000
0.079621 -0.392392 0.916346 0.000000
-0.996195 0.001369 0.087145 0.000000
10.772983 9.308582 -6.147681 1.000000
```

Figure 6.5: Example of .pos position file. The first 4 lines correspond to the aspect matrix, and the 4 last lines to the position matrix.

If more than one mesh are selected, the following message shows up :



6.3.2 Load transposed position

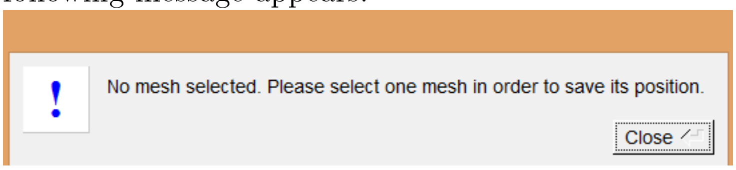
This option may be useful in the following case:

- Let us suppose that you did modify the position of a given surface and saved its position
- Then you have saved the surface in its current modified position (that is : the original position of the surface is lost).

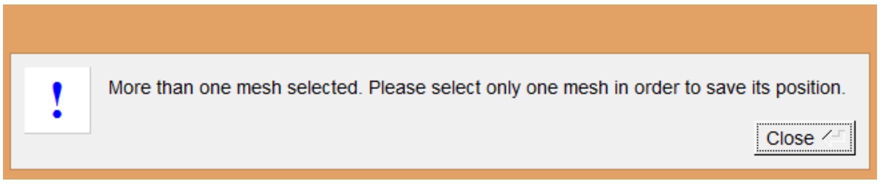
For some reason, you may need to open the surface in its original position. To do so, you may apply this option (apply transposed position matrix to the modified surface). Note : this option only works if the aspect matrix was not modified.

6.3.3 Save position

Mesh aspect and position matrices can be saved in “.pos" format. If no surface is selected, the following message appears:



If more than one mesh are selected, the following message shows up:



6.3.4 Edit manually aspect and position matrices

There are two ways to access the object matrix editor :

- Either select one surface and click on “"(edit first selected object position and aspect matrices).
- Or select one surface and click on “edit selected surfaces→Rendering modifications→ edit first selected object position and aspect matrices".

This opens the “Object Matrix" window, in which the aspect and position matrices can be edited. See “Edit selected surfaces" section 8.2.3 (Rendering modifications→ edit first selected object position and aspect matrices) for further information.

6.4 Project

When working with multiple surface objects, loading surfaces and associated positions one by one becomes fastidious. Besides, after having digitized landmarks, flags or having set tag names and colors, or having defined orientation labels, you may wish to open or save this information along with surface files. You may open and save series of meshes and associated position matrices, landmarks, flags, tag colors and labels and orientation labels using this menu. “Project" files (.ntw) files are organized the following way (see Fig. 6.6):
- Optional: name of orientation label file (.ori)
- Optional: name of tag file (.tag)
- Optional: name of flag file (.flg)
- Optional: name of landmak file (.lmk, .ver, .stv or .cur)
- Name of surface 1 file
- Name of position 1 file associated to surface 1
- Surface 1 RGB color and transparency
- Name of surface 2 file
- Name of position 2 file associated to surface 2
- Surface 2 RGB color and transparency (etc...)

Surface files can be of the following types : .stl, .vtk, .ply and .obj
“.ntw" files can be constructed manually, providing that the refered surface and position files exist.

```
exploded.ori
exploded.tag
exploded.flg
exploded.stv
Hippopotamus_amphibius_braincase.vtk
Hippopotamus_amphibius_braincase_exploded.pos
0.250980 0.482353 0.494118 1.000000
Hippopotamus_amphibius_bulla.vtk
Hippopotamus_amphibius_bulla_exploded.pos
0.647059 0.556863 0.086275 1.000000
Hippopotamus_amphibius_petrosal.vtk
Hippopotamus_amphibius_petrosal_exploded.pos
0.764706 0.356863 0.000000 1.000000
Hippopotamus_amphibius_sinus.vtk
Hippopotamus_amphibius_sinus_exploded.pos
0.470588 0.200000 0.568627 1.000000
```

Figure 6.6: Example of project .ntw file containing references to orientation labels, tags labels and colors, flags and landmarks files.

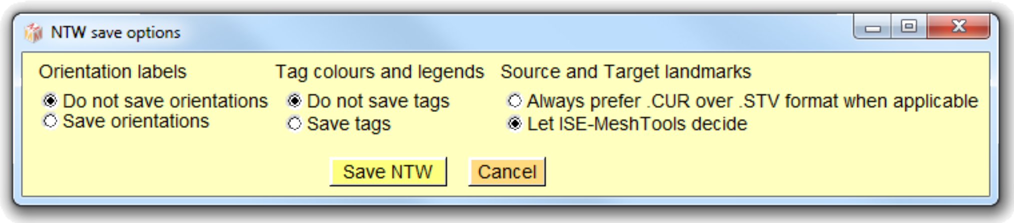


Figure 6.7: Save project window.

6.4.1 Open project

Loads a .ntw file

6.4.2 Save project

The Save project window (6.7) opens and asks whether :

- orientation labels should be saved along with the project (by default, orientation labels are not saved, as orientation labels are quite rarely set by users).
- tag colors and legends should by saved (by default, .tag file are not saved along with the project, as tag setting is not the most common task).
- to prefer .CUR over .STV format when applicable or to let ISE-MeshTools decide which landmark file format is best when saving projects. We advise you to let ISE-MeshTools decide unless you are in the process of digitizing curves (for instance on inner ears).

Furthermore, saving a .ntw file implies saving all selected surfaces. Though ISE-MeshTools can open .ntw files implicating .stl, .ply and .obj surfaces, when saving a .ntw project, surface files will be saved in .vtk format in order to keep potential tag / scalars associated to each saved surface. Each surface file will be given the name of the original file. Each position file will be given a name which starts with the name of the associated surface and ends with the name of the project. In the .ntw file example shown above, the surface files are

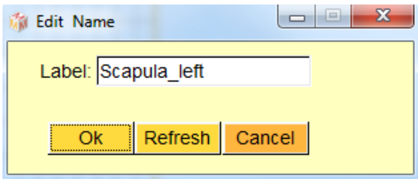


Figure 6.8: Edit name window.

- “Hippopotamus_amphibius_braincase.vtk"
- “Hippopotamus_amphibius_bulla.vtk"
- “Hippopotamus_amphibius_petrosal.vtk"
- “Hippopotamus_amphibius_sinus.vtk"

and the project name is “exploded.ntw". The advantage of naming position files that way is you may construct different .ntw files with different associated surface files using a same set of surfaces. Requirement : all selected surfaces saved via this option need to have distinct names. Note : When working with “project" files, you may need at some point to rename some of the object surfaces. To do so, select one surface, click on : the “Edit Name" window appears (see Fig. 6.8). Press ok to modify the name of that surface object. See tutorial “working with projects" for further information.

6.5 Landmarks

As mentioned earlier, landmarks can be set on surfaces by pressing “L" + left mouse click.

Two series ofconventional landmarks can be set : “normal" and “target" landmarks. As mentioned earlier, in the “normal" landmark mode (button active), pressing “L" + left mouse click results in the creation a “normal" landmark (a red one). In the “target" landmark mode (button active), pressing “L" + left mouse click will create a “target" landmark (a yellow one). “Normal" and “target" landmarks can be loaded and saved.
Selected “normal"/"target" landmarks can be reordered using the following buttons. Pressing “" will place the selected landmarks earlier in the “normal"/"target" landmark list, while pressing ““"" will place them one step further, respectively.

ISE-MeshTools can manage three types of landmark files: “.LMK", “.VER" and “.STV" files.

- .LMK files contain a series of lines, each line being constructed the following way (see Fig. 6.9): landmark name (without space or tab character), landmark coordinates. Note

```
landmark1: 6.179220 -1.516437 6.048982
landmark2: 5.260506 2.779373 6.386808
landmark3: 4.527496 4.214221 5.534292
```

Figure 6.9: Example of .LMK file

```
landmark1: 6.179220 -1.516437 6.048982 6.905368 -1.890056 6.626146
landmark2: 5.260506 2.779373 6.386808 5.862945 3.208343 7.059899
landmark3: 4.527496 4.214221 5.534292 5.015749 5.053222 5.774469
```

Figure 6.10: Example of .VER file

that each landmark name does not need to be of the form “landmark"+landmark number. Meanwhile, the name should not hold space or tab characters.

- .VER files contain a series of lines, each line being constructed the following way (see Fig. 6.10): landmark name (without space or tab character), landmark coordinates, landmark orientation.
- .STV files may contain one or two series of line. The first line contains two integers, the first being the type of landmark (0 for “normal" or 1 for “target"), the second being the number of lines of landmarks of this type which are expected to follow, constructed the following way: landmark name (without space or tab character), landmark coordinates, landmark orientation. An example of .STV file containing both “normal" and “target" landmarks is given in Fig. 6.11. Note that the number of “normal" and “target" landmarks saved within a .STV file can differ.

6.5.1 Load landmarks

Landmarks opened using this option will be put in the “normal" landmark list (red landmarks)

```
0 11
landmark1: 192.665222 1.212024 -3.060179 188.977325 1.315192 -1.845517
landmark2: -49.889797 1.817465 57.492958 -51.696289 1.995751 60.926819
landmark3: 71.462761 -52.624928 30.014076 73.076599 -52.861542 33.539150
landmark4: 70.316025 62.361145 29.645668 71.277863 62.512714 33.405788
landmark5: 67.744781 -35.594147 66.756393 68.846420 -37.872295 69.703094
landmark6: 68.661987 43.211151 65.035034 69.404091 46.795853 66.333420
landmark7: -2.099081 -46.220695 9.180318 -2.464601 -46.584351 5.330545
landmark8: -1.631284 50.896877 9.280211 -2.308331 50.677433 5.461828
landmark9: 156.577911 1.307144 26.860971 157.671432 4.085039 29.345751
landmark10: 74.154572 3.698074 -2.570488 73.504478 4.591600 -6.294141
landmark11: 34.070713 3.982986 5.395114 35.274883 3.865624 1.704206
1 6
landmark1: 0.000000 0.000000 0.000000 0.000000 0.000000 5.826221
landmark2: 0.000000 0.000000 254.000000 0.000000 0.000000 248.173782
landmark3: 1.000000 0.000000 0.000000 -3.119760 0.000000 4.119760
landmark4: 1.000000 0.000000 150.000000 0.960899 0.000000 144.173904
landmark5: 2.000000 0.000000 0.000000 -3.211130 0.000000 2.605565
landmark6: 2.000000 0.000000 70.000000 1.831195 0.000000 64.176224
```

Figure 6.11: Example of .STV file

6.5.2 Save landmarks

You may decide whether you wish to save only selected “normal" landmarks or all selected and unselected “normal" landmarks (the red ones).

6.5.3 Save target landmarks

You may decide whether you wish to save only selected “target" landmarks or all selected and unselected “target" landmarks (the red ones). The “Landmarks" chapter (chapter 9) and the tutorial “working with landmarks" contain further information regarding landmark digitization with ISE-MeshTools

6.5.4 Load source and target landmarks

Landmarks opened using this option will put in the “normal" and/or “target" landmark lists depending on how the .STV file is constructed.

6.5.5 Save source and target landmarks

All “normal" and“target" landmarks will be saved inside the .STV file.

6.6 Curves

3D Curves are constructed in ISE-MeshTools using 2 series of landmarks : a series of “normal" landmarks, and a series of “target" landmarks of equal sizes. “Target" landmarks are referred to as “curve handles", when they are used to construct curves (“Target" landmarks can also be used to achieve TPS deformation, see later in this documentation). By default, curves are not drawn in the main 3D window : curves start being drawn when the checkbox “draw curves" is checked in the menu “Viewing opt." (☑ Draw curves). Curves are draw green when no landmark/curve handle belonging to the curve is selected. Curves are drawn red when at least one landmark / curve handle involved in the curve is selected. Two different cases are considered:

- Case 1: the numbers of “normal" and “handle/target" landmarks differ. In that case, a curve is a series of lines passing through “normal" landmarks.
- Case 2: the numbers of “normal" and “handle/target" landmarks differ. In that case, a curve is a series of cubic Bezier curves passing through “normal" landmarks. For a given set of 2 “normal" consecutive landmarks (Ln and Ln+1) and their associated curve “handles" (Hn and Hn+1), a mirror image of Hn+1 relative to Ln+1 (H’n+1) is constructed. The Bezier curve involving Ln, Ln+1, Hn and Hn+1 starts from Ln, going toward Hn, and arrives at Ln+1 coming from the direction of H’n+1.

The explicit form of the curve is :

$$B(t) = (1-t)^3 Ln + 3(1-t)^3 tHn + 3(1-t)t^2 H'n + 1 + t^3 Ln + 1, t \in [0,1] \tag{6.1}$$

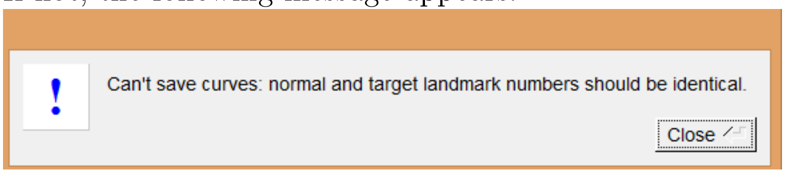
In order to be able to digitize several curves using a given set of normal and target landmarks, “normal" landmarks curves can be given 4 flags (see section 9.5 “Landmarks → Landmarks involved into curves" for further details):
Flag “0" : landmark is placed inside the curve (drawn “red").
Flag “1" : landmark is a curve start (drawn “green").
Flag “2" : landmark is placed inside the curve, and is a curve “milestone" (drawn blue) .
Flag “3" : landmark is placed inside the curve, and should be connected to the preceding curve starting point. When landmark n is flagged that way, landmarkn+1 will be set as a curve starting point.
Flag “2" is used to decompose a given curve into curve segments (see “export curves as landmark file"). By default, a curve comprises 1 segment
Flag “3" is used to close a curve (by default, curves are open).
3D curves are loaded and saved into .CUR files, which contain a series of lines, each line being constructed the following way: name (without space or tab character), curve “normal" landmark coordinates, curve “handle" coordinates, flag. In the example shown below, 4 curves are defined :
- an open curve starting from landmark 1 and ending at landmark 7
- a closed curve involving landmarks 8 to 12
- a closed curve involving landmarks 13 to 20
- a closed curve involving landmarks 21 to 26
These four curves contain only one segment (no curve milestone was set within those 4 curves). Note that each name does not need to be of the form “landmark"+ number. Meanwhile, the name should not hold space or tab characters.

6.6.1 Load curves

This menu allows the user to load a .CUR file.

6.6.2 Save curves

This menu allows the user to save current landmarks and curve handles as a .CUR file. This action is only allowed if the number of “normal" landmarks and “target" landmarks is the same. If not, the following message appears:



```
landmark1: 0.372506 -1.450796 -1.872275 0.096934 -2.083990 -2.584222 1
landmark2: 1.432226 -3.734149 -2.309953 2.264641 -4.044230 -2.146961 0
landmark3: 3.194311 -3.572493 -1.544599 3.531781 -3.243285 -1.326604 0
landmark4: 2.966238 -1.936326 -0.969882 2.507448 -1.342879 -1.036968 0
landmark5: 1.741511 -1.795793 -1.970134 1.545529 -2.004059 -2.316103 0
landmark6: 1.994205 -2.876585 -2.570605 2.372043 -3.069350 -2.591197 0
landmark7: 3.233635 -2.780582 -2.164102 3.420447 -2.374311 -1.791604 0
landmark8: 1.433686 0.362716 0.090703 1.815833 0.979808 -0.042806 1
landmark9: 0.459834 2.094835 -0.017995 -0.255083 2.359392 -0.043907 0
landmark10: -1.353654 1.515530 0.053875 -1.676283 1.001972 0.089944 0
landmark11: -1.196078 -0.347833 0.139561 -0.969538 -0.621288 0.193434 0
landmark12: -0.319761 -1.012853 0.301399 0.073680 -1.063024 0.320989 3
landmark13: 2.309960 0.220924 0.770375 2.547965 0.511319 0.738723 1
landmark14: 3.079090 1.135534 1.290322 3.155088 1.323655 1.833658 0
landmark15: 2.687151 1.154598 3.467043 2.201378 0.939847 4.014127 0
landmark16: 1.161453 0.210946 4.287964 0.035428 -0.379038 4.438488 0
landmark17: -0.450655 -1.069932 2.840355 -0.508689 -1.210229 2.558033 0
landmark18: -0.650342 -1.268928 2.137542 -0.593531 -1.326798 1.771273 0
landmark19: -0.122183 -1.404526 0.722653 0.036526 -1.383950 0.581831 0
landmark20: 1.079893 -0.725003 0.706101 1.364071 -0.583780 0.706101 3
landmark21: -1.060005 -1.281700 -0.429272 -1.154971 -1.147840 -0.539899 1
landmark22: -2.110059 -0.179460 -0.926232 -2.666639 0.729053 -0.718993 0
landmark23: -2.796323 0.990297 0.987866 -2.730658 1.079654 1.956513 0
landmark24: -1.831005 0.069832 2.528836 -1.428311 -0.362153 2.749845 0
landmark25: -0.703198 -1.255077 2.136627 -0.703198 -1.255077 2.136627 0
landmark26: -0.174492 -1.394222 0.706781 -0.244929 -1.403785 0.448291 3
```

Figure 6.12: Example of .CUR file

If you are in the process of digitizing curves (and do not have achieved to digitize the same number of “normal" and “target" landmarks and wish to save the current state of your work, you may decide to save a .STV file (see section 6.5.5).

6.6.3 Export curves as landmark file

Curves can be transformed in a series of equidistant landmarks using this option. The curve decimation window appears. Each curve/curve segment is saved as a number of equidistant landmarks. In the present example, each curve/ curve segment is saved as 20 equidistant landmarks.

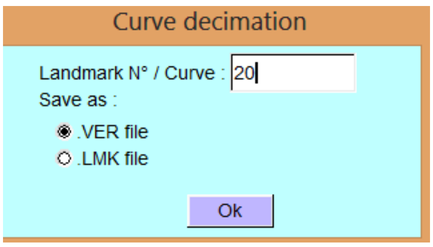
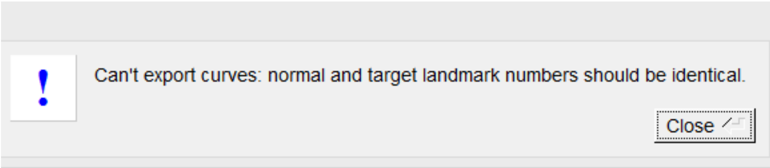


Figure 6.13: Curve decimation window

When pressing “Ok", if the numbers of “normal" landmarks and “target" landmarks differ, the following message appears:



6.6.4 Save curve infos (length per curve ...)

Each curve /curve segment length can be saved as a .txt file using this option. The “Landmarks" chapter (chapter 9) and the tutorial “Working with curves" contain further important information regarding curve digitization with ISE-MeshTools.

```
Curve 1 : length= 11.464937 mm
Curve 2 : length= 9.685019 mm
Curve 3 : length= 13.734188 mm
Curve 4 : length= 11.330129 mm
```

Figure 6.14: Example of curve info file

6.7 Tags and flags

- Tag colors and names can be edited interactively by clicking on , which opens the tag window (see the chapter “Tags" (chapter 11) and the tutorial “Working with tags" for further information).
- By default, Tags are not visible. To activate/deactivate tag display, click on
- Using “Tag display mode" () is useful when editing surface tags.
25 Tag names and associated colors can be defined in ISEMeshTools. Tag colors files (.TAG) consist of 25 pairs lines, each pair being constructed following way :
line 2*n: Tag name
line 2*n+1: Tag color and transparency

```
exterior
0.658824 0.560784 0.372549  1.000000
cochlea
0.000000 0.552941 0.686275  1.000000
lateral canal
0.000000 0.000000 1.000000  1.000000
anterior canal
0.556863 0.031373 0.474510  1.000000
posterior canal
0.827451 0.713726 0.117647  1.000000
vestibule
0.388235 0.125490 0.800000  1.000000
ovale window
0.000000 0.521569 0.145098  1.000000
round window
1.000000 0.474510 0.000000  1.000000
common crus
0.290196 0.521569 0.000000  1.000000

(etc...)
```

Figure 6.15: Example of .TAG file

Regarding flags, as stated earlier, one series of “flag landmarks" can be set in ISE-MeshTools (button should be pressed). To edit flag label, length and color, select one flag landmark, click on . The “edit flag" window appears. Pressing ok will update the label, the color and the length associated to the selected flag, which in turn will be unselected. If you wish to edit a second flag, select it and press “Refresh". The current color, length and label of the newly selected flag will appear in the edit flag window.

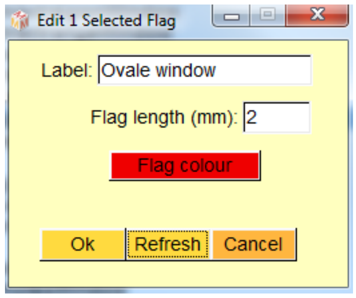


Figure 6.16: Edit Flag window

Flags are saved using the .FLG file format, which consists of n pairs of lines constructed the following way :
line 2*n: Flag name
line 2*n+1: Flag coordinates, flag orientation, flag length and color.

6.7.1 Load tag colors and labels

Select a .TAG file using this menu → Then open the tag window ( ) : Tag labels, colors and transparencies should have been updated.

6.7.2 Save tag colors and labels

This option saves the current state of tag labels, colors and transparencies in a .TAG file.

6.7.3 Load flags

Select a .FLG file using this menu

6.7.4 Save flags

This option saves the current flag landmarks into a .FLG file, regardless their selection status.

6.8 Save infos (surface area, volume...)

Surface area, volume, triangle number and vertex number of selected surface objects can be saved in a .txt file using this option.

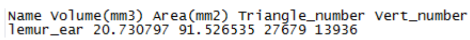

Name Volume(mm3) Area(mm2) Triangle_number Vert_number
lemur_ear 20.730797 91.526535 27679 13936

Figure 6.17: Example of info file

Note: surface objects should be closed in order to provide a correct estimation of object volume.

6.9 Orientation labels

The coordinate system orientation helper labels can be saved into ".ORI" files, which are .txt files containing 6 lines, 1 for each axis.

6.9.1 Load orientation labels

Select a .ORI file using this menu → Then open the orientation labels window window (Viewing opt; Orientation labels) : the 6 orientation labels should have been updated.

6.9.2 Save orientation labels

This option saves the current state of orientation labels in a .ORI file.

Chapter 7

Viewing Options

Contents

7.1 General color and lightning options

The general options window contains the following sections:

7.1.1 Windows

These controls affect the default color of objects opened within ISE-MeshTools, the color of grid elements, and the background color.

7.1.2 Light

These controls affect the orientation of the light and specular, diffuse and ambient light parameters. By default surface back faces are not shown. Back face lighting can be enabled if the checkbox “enable two sided lighting" is checked

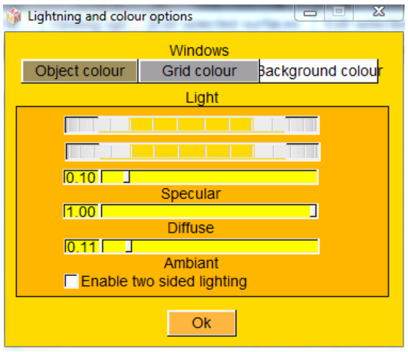


Figure 7.1: Lightning and color options window

7.2 General rendering options

The rendering options window contains the following sections:

7.2.1 Object rendering when moving object/camera

Show full surface / scalars (slower): when active, surfaces are fully drawn when moving the object or the camera. This results in a better perception of object / camera movements. This option is convenient when working with light surfaces.

Show point cloud (faster): 3D surface rendering can be slow using the preceding option when working with:
- large number of surfaces simultaneously
- heavy surfaces (large number of triangle / vertices)

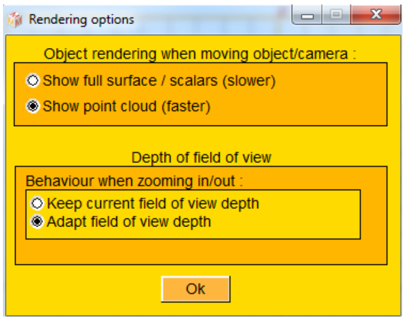


Figure 7.2: Rendering options window

Also, rendering is slower when tag rendering mode is active ( ) or when scalar rendering mode is active ( ) . In order to increase rendering speed, surfaces can be rendered as a schematic point cloud when moving the object or the camera.

7.2.2 Depth of field of view

When the option “Adapt field of view depth" is active, changing the zoom value will affect the depth of the field of view (camera.far value) and the position of the clipping plane (camera.tz

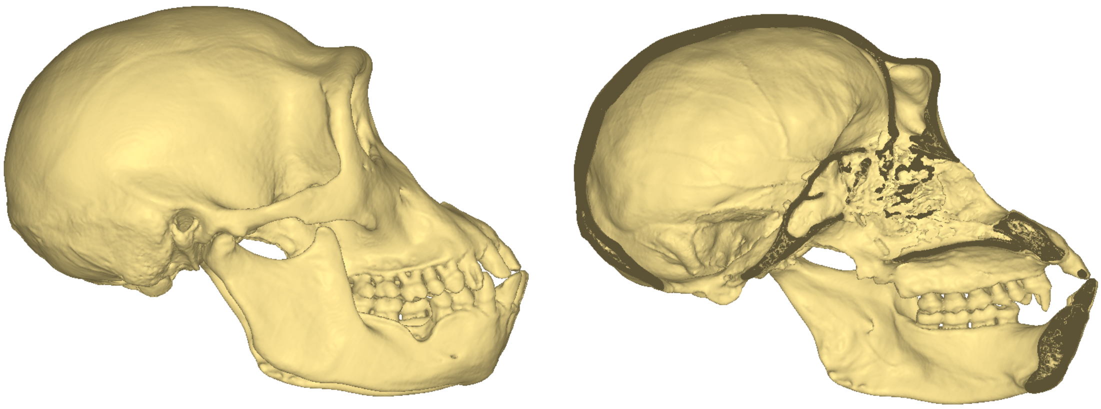


Figure 7.3: Left : cranium and mandible of the type specimen of *Pan paniscus*(downloadable at http://www.metafro.be/primates/panpaniscustype) rendered with the clipping plane placed at its original position. Right : the same skull viewed using a clipping plane placed at the position of the sagittal plane, revealing inner structures such as the endocranium cavity. Note that the position of the specimen was modified so that the sagittal plane of the skull passes through the point (0,0,0) and is perpendicular to the vector (0,1,0).

value). When the option “Keep current field of view depth" is active, changing the zoom will not affect the camera.far and camera.tz values.

7.3 Camera

7.3.1 Camera options

- Camera near, Camera far : define near and far clipping - planes of the camera
- Azimuth, Elevation, Twist : camera rotation parameters
- Tx, Ty, Tz : camera translation in x, y and z

The buttons  and  which lie underneath the Tz control(and also underneath the clipping plane slider in the main window) permit to adjust / readjust the position of the clipping plane at predefined positions :

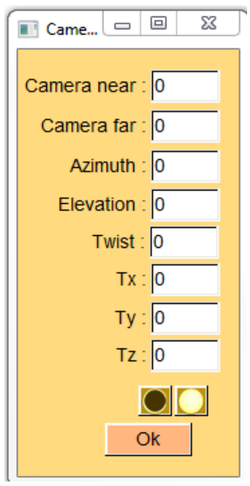


Figure 7.4: Camera options window

- : the clipping plane is placed at z = 0 (all objects having a z coordinate along z viewing

axis smaller than 0 are hidden).
- : the clipping plane is replaced at its original value : z= - camera.far / 2. This value permits to view objects having positive and negative coordinates along z viewing axis.

7.3.2 Camera rotation centre at

Camera rotation centre can be set at the origin of the coordinate system (x=0, y=0, z=0) or at the location of one of the first 10 “normal" landmarks.

7.3.3 Set 100 pixels in mm

Camera zoom can be modified in order to reach a desired display pixel size. This is useful to produce scaled images.

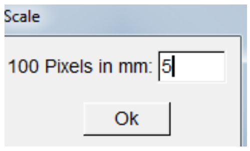


Figure 7.5: Scale window

7.3.4 Reset camera

You may use this option to reinitialize camera parameters.

7.4 Object rendering options

7.4.1 Gouraud shading

This is the default rendering mode. Object rendering is performed using vertices’ normals.

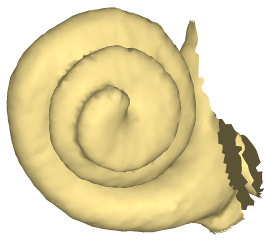


Figure 7.6: Gouraud shading

7.4.2 Draw wireframe

This option can be useful to inspect the structure of the surface.

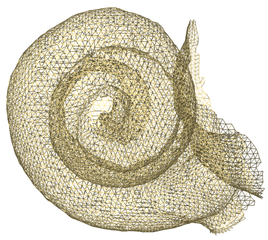


Figure 7.7: Example of wireframe rendering

7.4.3 Sort vertices from back to front

This option is (only) useful when working with transparent surfaces, in the case you want to display all surface inner structures. When active, whenever the camera of the object is moved, vertices will be sorted in order to be displayed from back to front. This is the way transparency is achieved in OpenGL. As a consequence, rendering fluidity becomes slow when working with heavy surfaces and/or a large number of objects. To change the transparency of an object, select an object, and reach "Edit selected surfaces → Rendering modifications → Set alpha value".

Alternatively, in order to increase rendering fluidity, you may chose not to use the "Sort vertices from back to front beware: slow rendering)" rendering option, but to sort episodically the vertices from back to front by pressing the button .
Working with transparency is useful to observe inner structures and/or when digitizing landmarks inside structures.

Figure 7.8: Sort vertices from back to front.In this example, vertices are displayed from back to front. Four landmarks placed within this structure can be visualized.

7.4.4 Sort vertices from front to back

This option is (only) useful when working with transparent surfaces, and when you do not want to display too many inner structures. When active, whenever the camera of the object is moved, vertices will be sorted in order to be displayed from front to back. As a consequence, rendering fluidity becomes slow when working with heavy surfaces and/or a large number of objects. Alternatively, in order to increase rendering fluidity, you may chose not to use the "Sort vertices from front to back (beware: slow rendering)" rendering option, but to sort episodically the vertices from front to back by pressing the button .

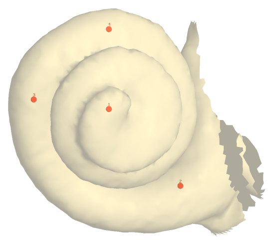


Figure 7.9: Sort vertices from front to back.

7.4.5 Flat triangles

Using this option improves the perception of surface structure.

Figure 7.10: Example of flat triangle surface rendering

7.4.6 Wireframe and flat triangles

Using this option further improves the perception of surface structure.

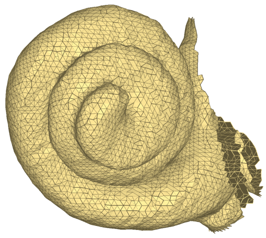


Figure 7.11: Example of flat triangle + wireframe surface rendering

7.4.7 Backface culling

Surface's back faces are hidden when using this option.

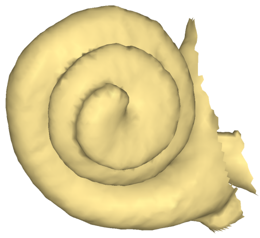


Figure 7.12: Backface culling rendering

7.4.8 Display vertices ids

For surface inspection purposes, you may sometimes need to visualize vertices ids. Note : this option affects rendering fluidity even when using relatively light surfaces.

7.4.9 Display triangle ids

For surface inspection purposes, you may sometimes need to visualize triangle ids.
Note : this option affects rendering fluidity even when using relatively light surfaces.

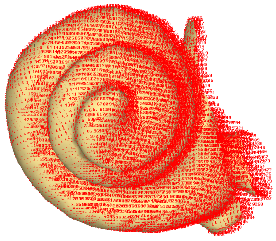


Figure 7.13: Gouraud shading + vertices ids rendering

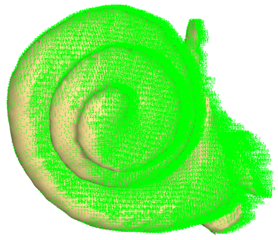


Figure 7.14: Gouraud shading + triangle ids rendering

7.5 Grid size

Grid rendering can be edited to reach the desired size/square

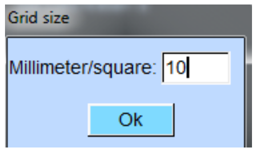


Figure 7.15: Grid size window

7.6 Landmark and flag rendering options

The landmark and flag options window (see Fig. 7.16) contains the following sections:

7.6.1 Landmarks rendering

“Normal" and “Target" landmarks can be drawn as spheres or as needles. Landmark display size can be chosen to be adjusted automatically (default behaviour) or edited manually (when you feel that the automatic adjustment does not meet your needs) also edited manually in this section.

7.6.2 Flags settings

Flag length and color settings can be defined in thissubsection. Once placed on a surface and selected, the color, the length and the label of the flag can be changed by pressing .

7.7 Display landmark numbers

“Normal" and “target" landmark numbers are displayed by default. For illustration purposes, you may sometimes need to hide landmark numbers.

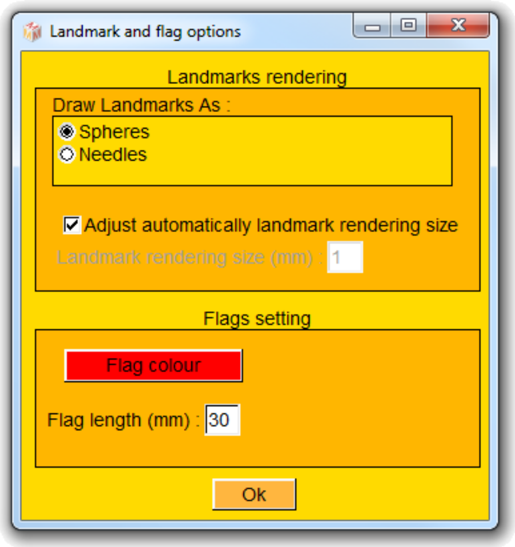


Figure 7.16: Landmark and flag options window.

7.8 Draw curves

Activate/deactivate this option to draw/hide 3D Bezier curves passing through "normal" landmarks. See the tutorial "working with curves" for further details regarding curve digitization with ISEMeshTools.

7.9 Orientation labels

The labels associated to the 6 orientation axes can be modified using this window. This will affect the coordinate system orientation helper displayed on the bottom left area of the 3D window.

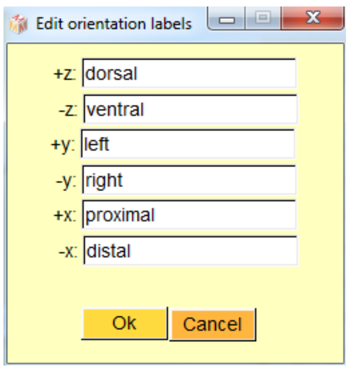


Figure 7.17: Edit orientation labels window

7.10 VBO activate (GPU acceleration)

By default, ISE-MeshTools does not use the graphic card to display 3D objects, which slows down dramatically the rendering speed of heavy 3D meshes and that of tagged meshes. However, we have chosen not to activate the "VBO" rendering option by default, as we have experienced that it causes ISE-MeshTools to crash on old computers or on some others which have not up to date graphic card drivers. By activating the graphic card acceleration option (VBO activate), you will probably find out that ISE-MeshTools is much more fluid to work with.
This feature was implemented by Cécile Peladan.

Chapter 8

Edit selected surfaces

Contents

As a prerequisite, you need to select surfaces. Only selected surfaces can be edited.

8.1 Structure modifications

8.1.1 Invert

A given surface's triangles can be inverted in order to show inner structures (see Fig. 8.1).

Note that the original surface is directly affected; this option does not involve the use of a filter (no output additional surface is created).

8.1.2 Mirror

This option uses vtkReflectionFilter,which produces a mirror image of the original selected input mesh (see Fig. 8.2).

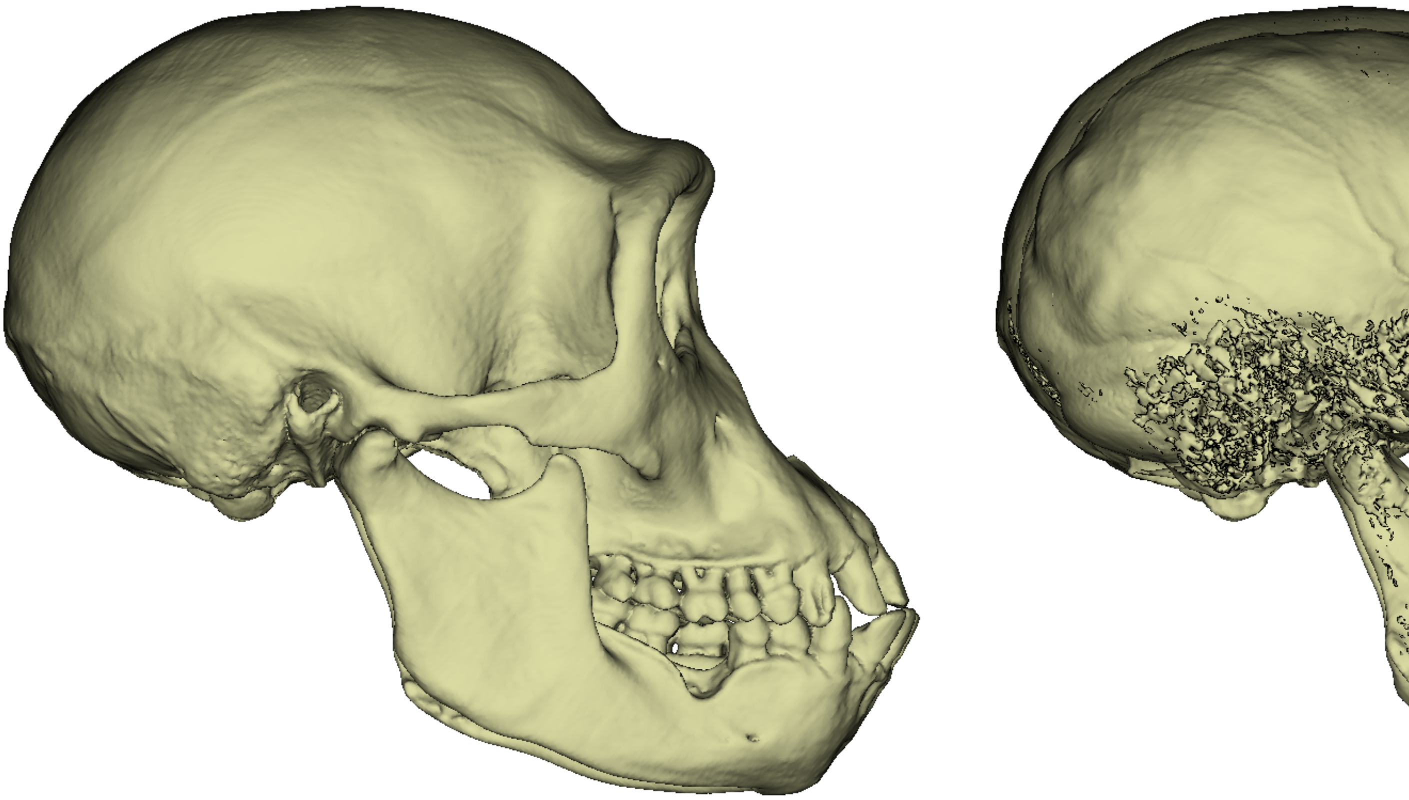


Figure 8.1: Example of surface triangle inversion. Left: original surface. Right: the same surface inverted, revealing inner structures such as the endocranial cavity. Gouraud shading rendering + backface culling option was used.

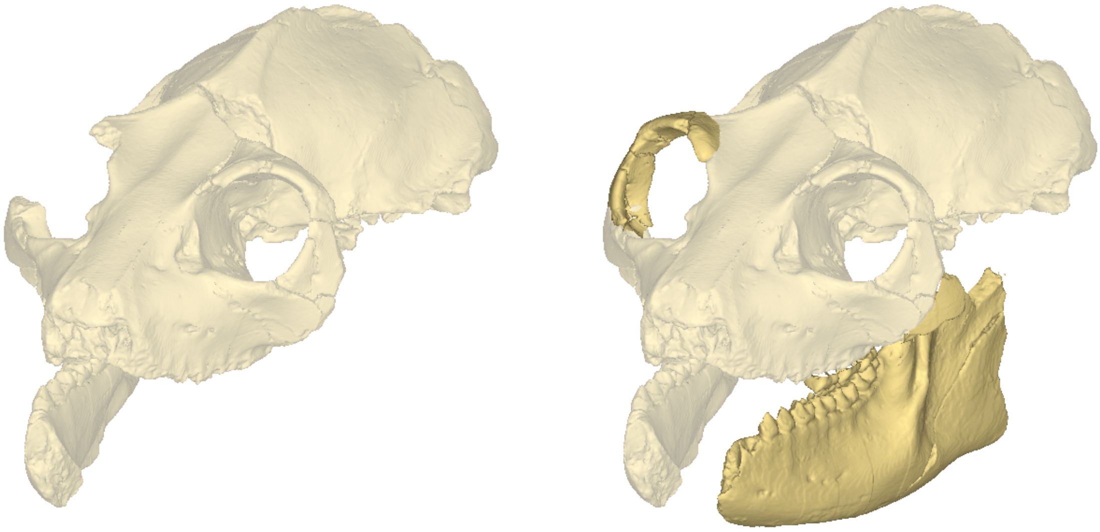


Figure 8.2: Example of fossil restoration implicating the production of mirror images of missing parts.

8.1.3 Connectivity : separate all non connected regions

This option uses vtkPolyDataConnectivityFilter. This filter produces a new surface for each non-connected region of the selected input surface. 3D meshes of biological objects sometimes contain a multitude of small and biologically irrelevant independent regions. This “noise" may have multiple origins: low quality of original 3D data, state of preservation of the specimen, threshold used to be able to visualize all relevant structures, etc... In order to extract relevant independent regions, only regions reaching a minimal size (minimal number of triangles) are transformed into new surfaces (see Fig. 8.4). This process may take some time to be completed. All produced surfaces corresponding to independent regions can be manipulated independently (see Fig. 8.5).

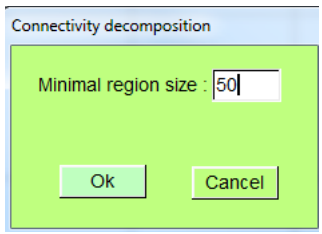


Figure 8.3: Connectivity decomposition window

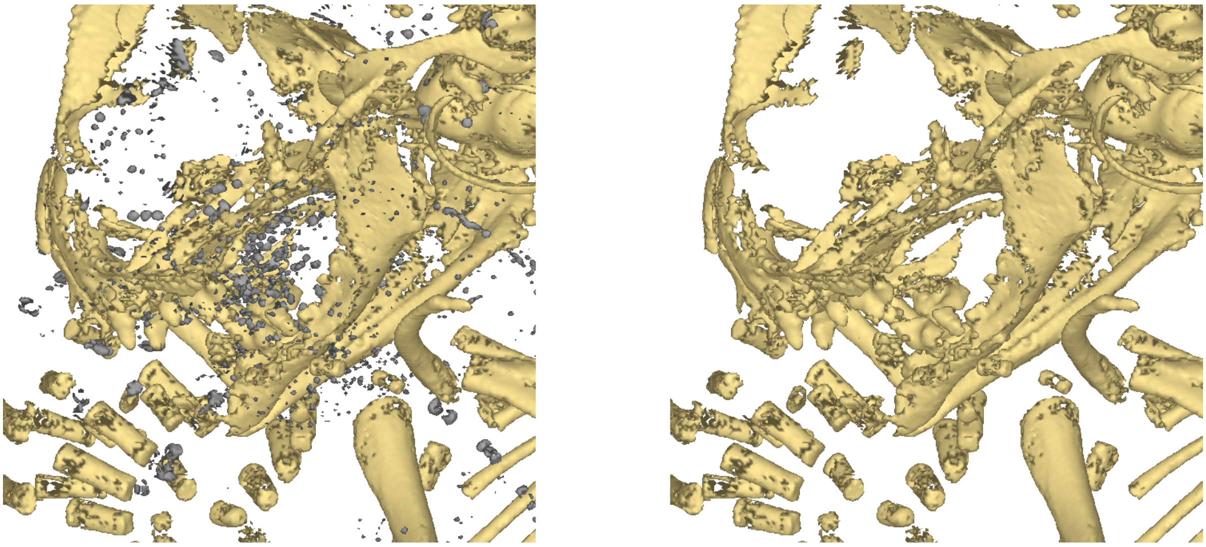


Figure 8.4: Left: Original surface containing a large number of independent regions of size greater than 350 triangles (in grey). Right : Filtered surfaces. All meshes produced using this filter have more than 350 triangles.

8.1.4 Connectivity: keep largest region

This option uses vtkPolyDataConnectivityFilter.This filter produces a new surface for the largest independent region of the selected input surface (see Fig. 8.6).

8.1.5 Lasso cut

You may cut through an input selected surface using this option (see Fig. 8.7). Once “lasso cut" menu is clicked, the mouse cursor changes to a cross in the 3D window. Additional mouse and keyboard controls become available (see Table 8.1).

Once “Middle click" is pressed or “C" + right click is pressed, the usual mouse and keyboard controls become available again.

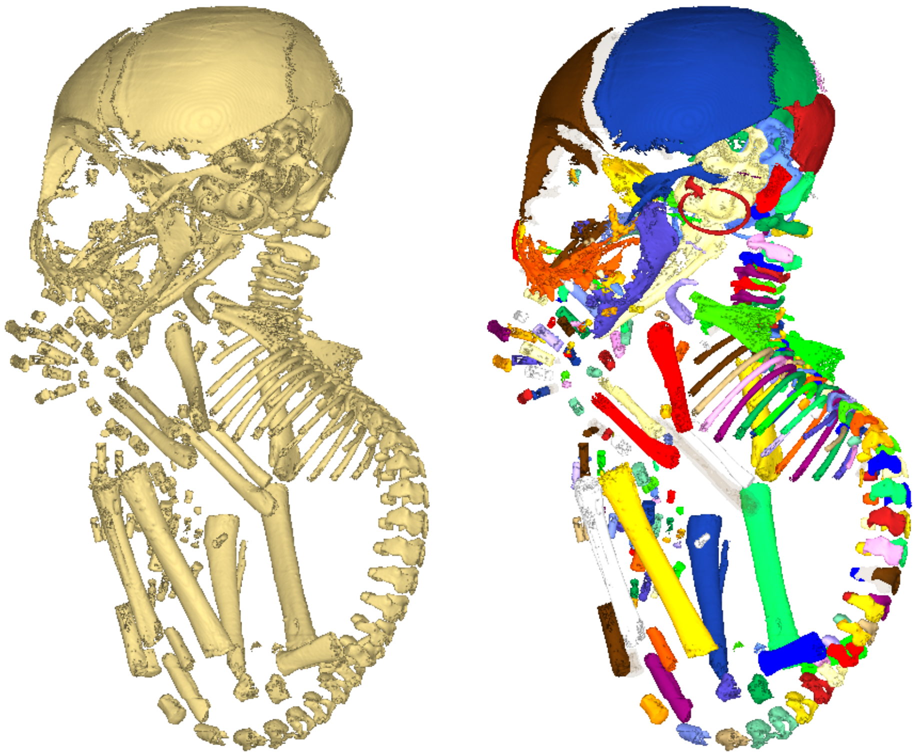


Figure 8.5: Left: original surface (one single mesh). Right: the resulting 298 filter output surfaces (each drawn using a different color) can be manipulated independently.

Left click	Adds a segment to polygon (segments are drawn yellow)
Right click	Connects last segment to first segment. If two segments cross each other, lasso action is cancelled. Otherwise, the closed polygon is drawn red.
Middle click or "C" + right click.	Once the lasso is closed (lasso polygon drawn red after a right click): when the click falls **inside**/*outside* the closed red polygon: the region falling <u>inside</u> the polygon **is included**/*is not included* into the filter output surface, respectively. The region falling <u>outside</u> the polygon is **is not included**/*is included* inside the output, respectively.

Table 8.1: Lasso cut controls

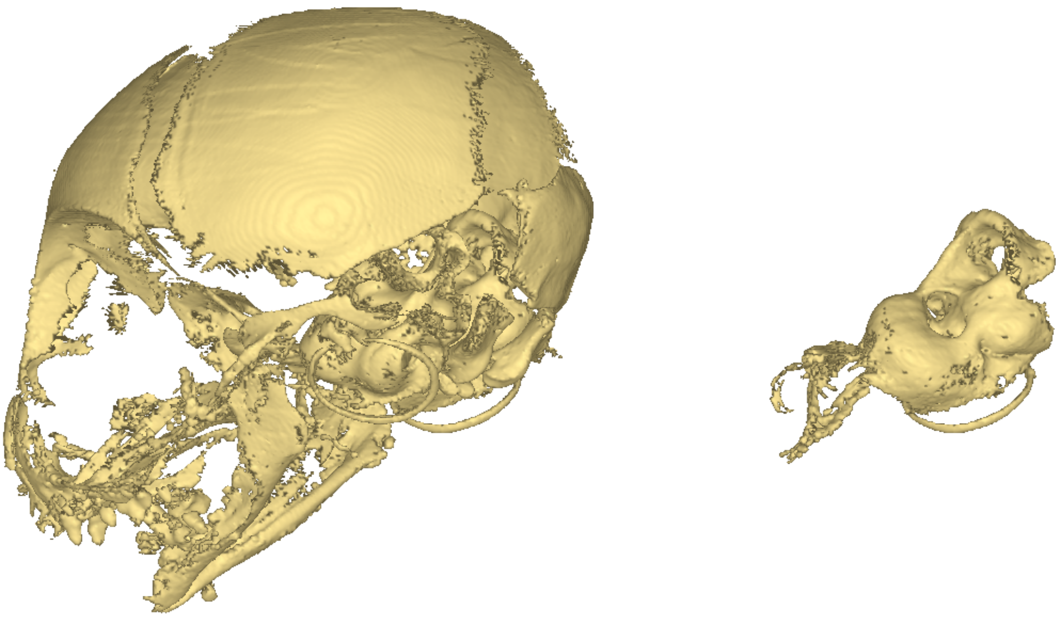


Figure 8.6: Left: original surface. Right: the resulting largest region in terms of triangle number.

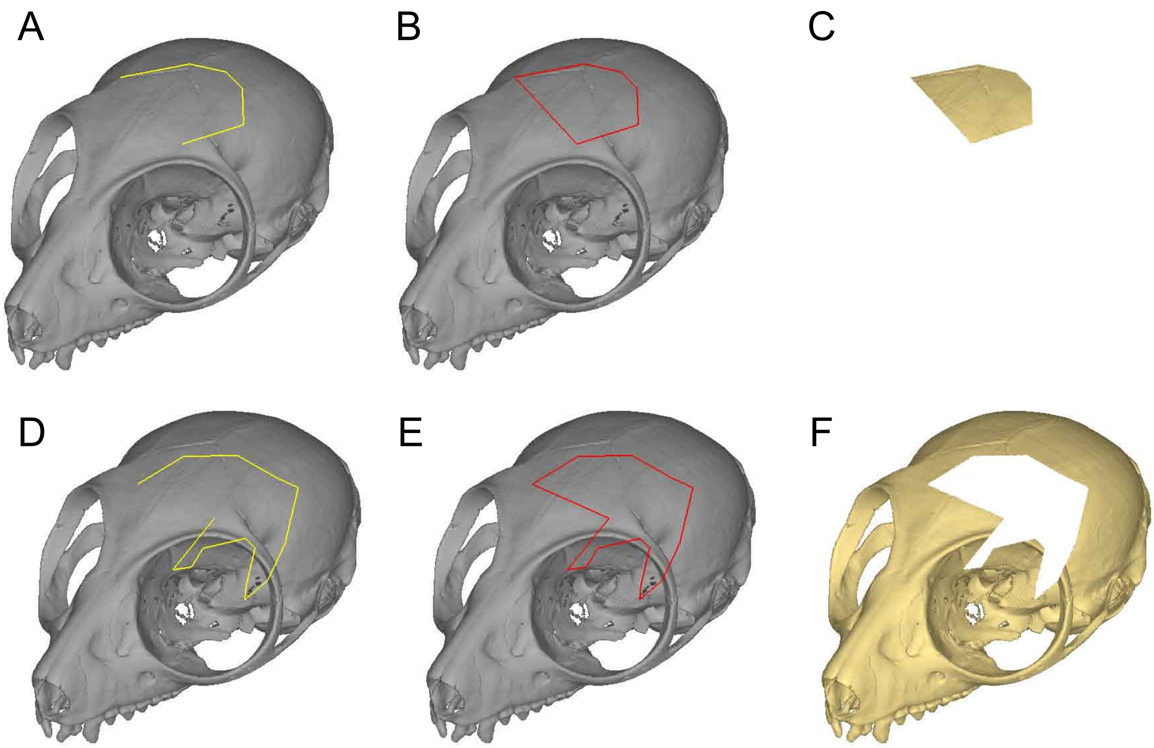


Figure 8.7: Top row. A: Left clicks → add new yellow segments. B: Right final click → connects last segment to first segment. C: Result of middle click inside red polygon. Bottom row. C: Left clicks → add new yellow segments. D: Right final click → connects last segment to first segment. E: Result of middle click outside red polygon.

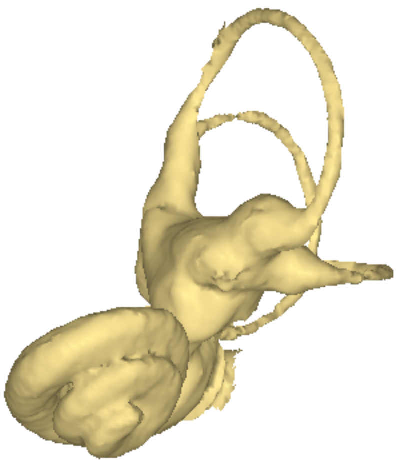


Figure 8.9: Left: example of original input surface. Right: resulting output surface after 50 iterations using a relaxation factor of 0.1.

8.1.6 Smooth

This option uses vtkSmoothPolyDataFilter.You may smooth an input selected surface using this option (see Fig. 8.9). A number of iteration and a relaxation factor are required.
See vtkSmoothPolyDataFilter documentation for further information regarding this option.

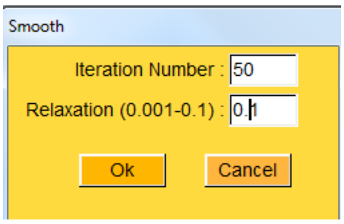


Figure 8.8: Smoothing window

8.1.7 TPS deformation

This option uses vtkThinPlateSplineTransform filter. Requirements : to use TPS deformation, a selected surface, a series of “n" normal landmarks and a series of “n" target landmarks (n>3) are needed. “Normal" landmarks are usually placed on the original selected input surface, whereas “target" landmarks are placed at a location in 3D space which will drive the TPS deformation (see Fig. 8.11). See vtkThinPlateSplineTransform documentation for further information regarding TPS deformation.

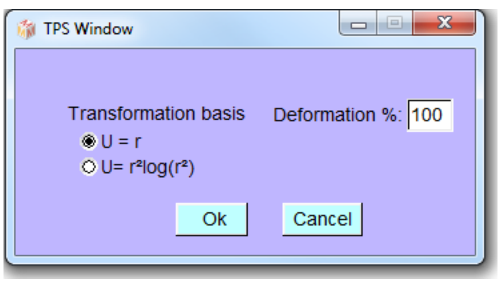


Figure 8.10: TPS window

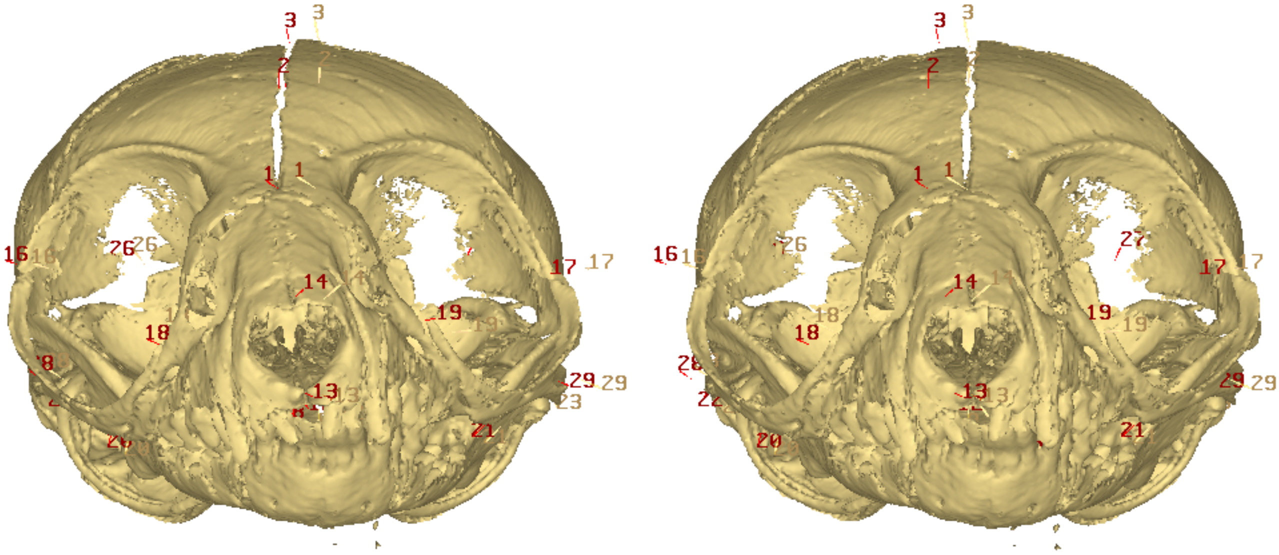


Figure 8.11: Left: original distorted input surface. 31 “normal" landmarks were placed on the surface and 31 “target" landmarks were placed in order to restore bilateral symmetry. Right: resulting output (deformation : 100%). Note that the 31 “target" landmarks are located on the output surface.

8.1.8 Decimate

This option uses vtkDecimatePro and vtkQuadricDecimation filters. Requirements : to use mesh decimation, a selected surface is required (see for instance Fig. 8.13). See vtkDecimatePro and vtkQuadricDecimation documentations for further information regarding mesh decimation.

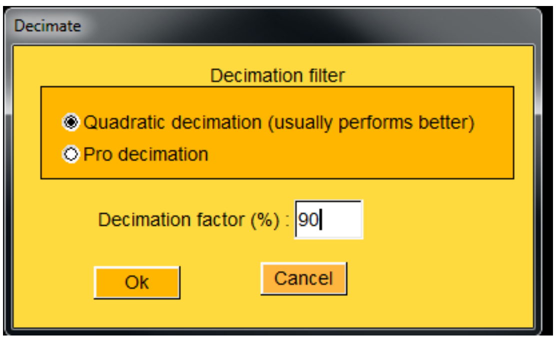


Figure 8.12: Decimate window

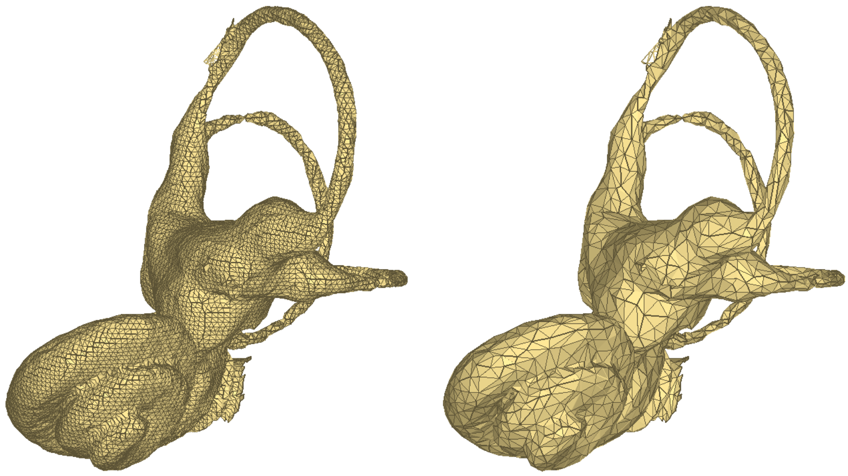


Figure 8.13: Mesh decimation. Left: original input surface. Number of triangles: 27679. Right: resulting output (vtkQuadricDecimation filter, decimation factor: 80%). Number of triangles:5535.

8.1.9 Densify

This option uses vtkDensifyPolyData filter. Requirements : to use mesh densification, a selected surface is required (see for instance Fig. 8.15). Note that mesh decimation can become extremely slow when using number of subdivisions larger than 1. See vtkDensifyPolyData documentation for further information regarding mesh densification.

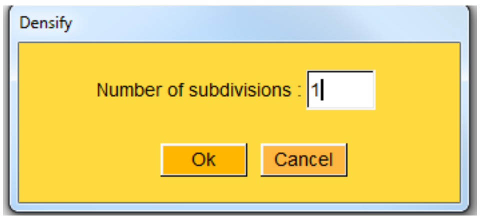


Figure 8.14: Densify window

8.1.10 Fill holes

This option uses vtkFillHolesFilter. Requirements : to use mesh hole filling, a selected surface is required (see for instance Fig. 8.17). See vtkFillHolesFilter documentation for further information regarding hole filling.

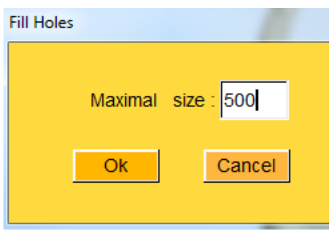


Figure 8.16: Fill holes window

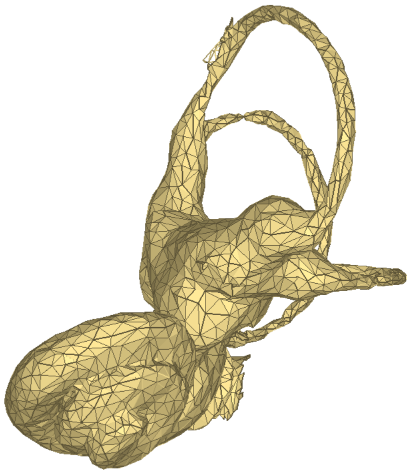


Figure 8.15: Mesh densification. Left: original input surface. Number of triangles: 5535. Right: resulting output (number of subdivisions: 1). Number of triangles: 16605.

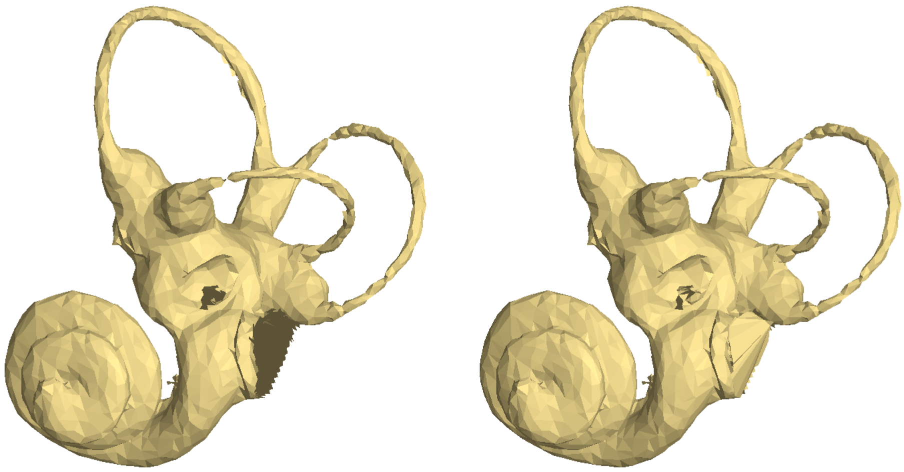


Figure 8.17: Filling holes. Left: original input surface. Number of triangles: 5535. Right: resulting output (maximal size: 1). Number of triangles: 5689.

8.1.11 Registration ICP

This option was implemented by Cécile Peladan, and deserves a chapter of its own and a tutorial, which will be written in a near future.

8.2 Rendering modifications.

8.2.1 Set alpha value

A selected surface is needed.
Please chose a value between 0 and 100. 100 stands for “opaque rendering". 0 stands for “invisible surface".

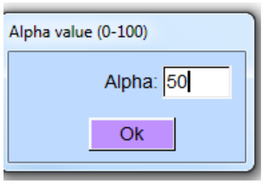


Figure 8.18: Alpha value setting window

As stated earlier, vertex display order has consequences on 3D rendering when working with transparency. 3D objects are displayed one after the other following an object list. The way object are ordered inside this list thus affects transparency rendering, especially when some objects such as inner structures are positioned inside others. Object display order of selected objects can be changed using the two following controls. Pressing “▲" will place all selected objects one step earlier in the object display list. Pressing “▼" will place all selected objects one step further in the object display list (see for instance Fig. 8.19).

See also “Sort vertices from back to front (beware: slow rendering)" (section 7.4.3) and “Sort vertices from front to back (beware: slow rendering)" (section 7.4.4) for further information.

8.2.2 Change object color

Object color can be changed using this option. A set of 13 predefined colors is available via this menu. Alternatively, you can edit object color manually using the “Object color" control of the “General options" window (click on menu “Show →General options→").

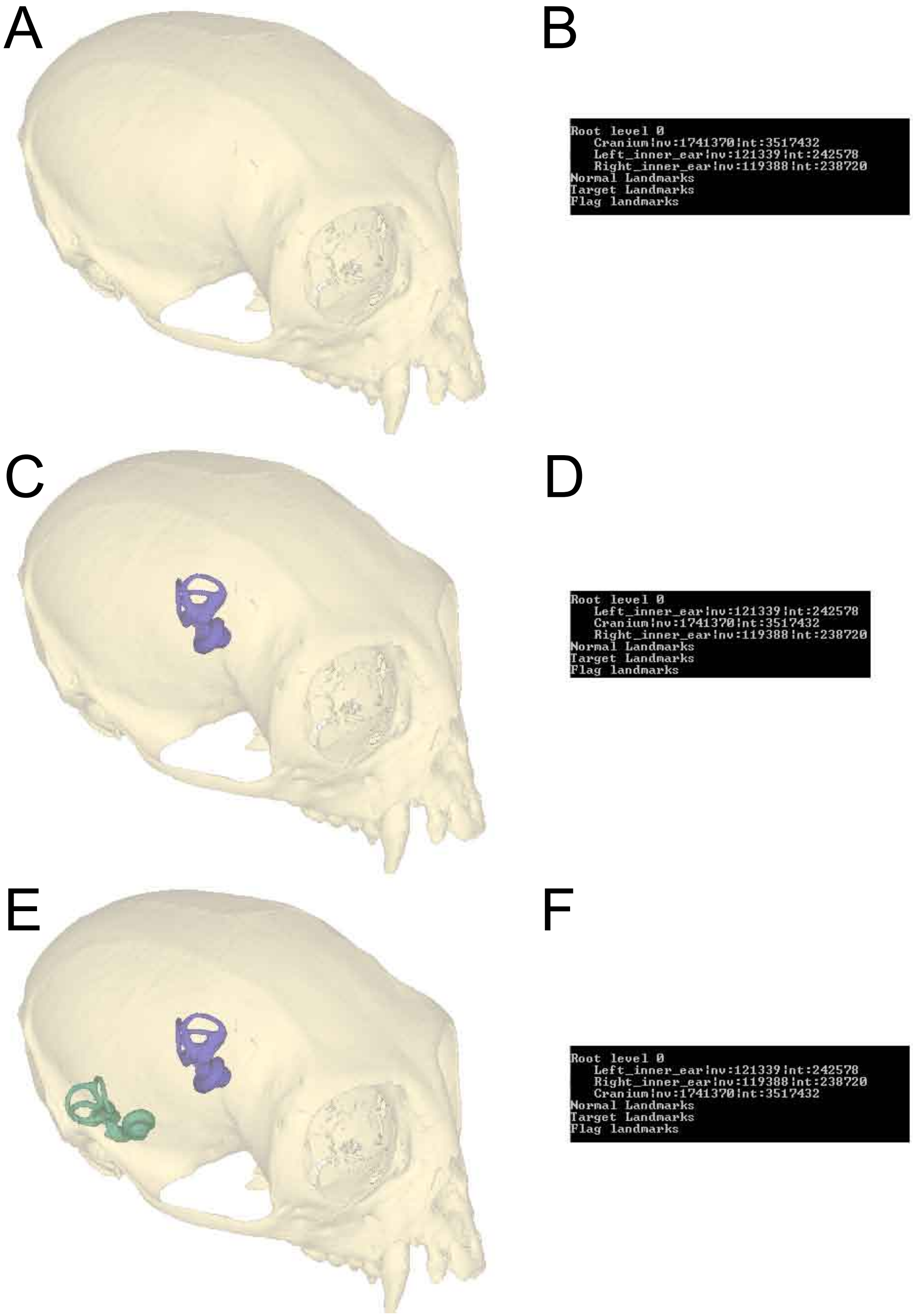


Figure 8.19: Transparency rendering. A. 3 objects (cranium, left_inner_ear and right_inner_ear) are opened in this order, and the object "cranium" is rendered with an alpha value of 40. The inner ears remain invisible, because "cranium" is displayed before the two inner ears are. B: corresponding display object order (Show →Display object order). C: the object "Cranium" was selected and put downward in the object display list. Now the left inner ear is visible, as it is displayed before the cranium is. D: corresponding display object order. E: the object "Cranium" was selected again, and put downward in the object display list. Now the two inner ears are visible, because they are both rendered before the cranium is. F: corresponding display object order.

8.2.3 Edit first selected object and aspect matrices

A selected surface is required . This opens the following window, in which the aspect and position matrices can be edited. Options :
- Ok : set aspect and position matrices to first selected object.
- Init : set aspect and position matrices to identity.
- Refresh : change matrices to those of first selected object.
- Ok for all selected objects: set aspect and position matrices to all selected objects.
You can also access faster the “Object Matrix" window by clicking on “"(edit first selected object position and aspect matrices).

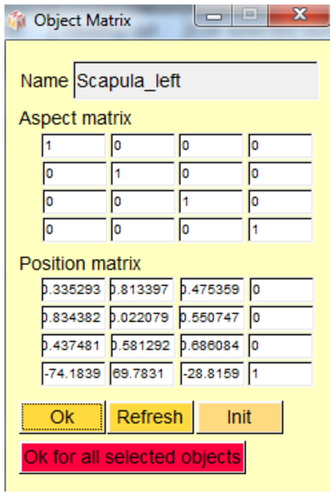


Figure 8.20: Object matrix window

8.3 Grouping actions.

One or several meshes can be placed into one logical object using this option. This option is useful in the following two cases.
- It allows an easy way to select/unselect together several objects (sometimes you do not want to select/unselect manually multiple meshes one by one).
- As a logical object's aspect object and position matrices can be edited, placing 1 or several meshes into one logical object provides a convenient means to achieve deformation into a particular direction in 3D space (for instance to achieve mesh decompression in one direction, see for instance Fig. 8.21).
Logical objects can be also put into another logical object. Note that meshes contained into selected logical objects cannot be saved until they are ungrouped.

8.3.1 Group

Select one or several meshes. Click on “Edit selected surfaces→Grouping actions→Group". As a result, the selected meshes are drawn in brown color.

8.3.2 Ungroup

Select one logical object. Click on “Edit selected surfaces→Grouping actions→Ungroup". If the logical object aspect and position matrices are different from the identity matrices, the position and aspect matrices of the contained objects are edited in order to take into account position and aspect transformations that were applied to that logical object. Then the objects contained into the selected logical object are ungrouped, and the selected logical object is deleted.

8.4 Object list order.

As explained earlier, the following options are useful when working with transparent objects. Normal and Target landmark ordering can also be edited using these options.
As stated above, access to object display order can be reached via the menu "Show→Display object order".

8.4.1 Move up

All selected objects (landmarks or surfaces) will be placed one step earlier in the object display list. This action can also be reached via the following button:

8.4.2 Move down

All selected objects (landmarks or surfaces) will be placed one step further in the object display list. This action can also be reached via the following button: See tutorial "working with landmarks" for further information.

8.5 Delete small objects.

8.5.1 Threshold : a given number of triangles

Using this option, you may delete selected objects smaller than a given number of triangles.

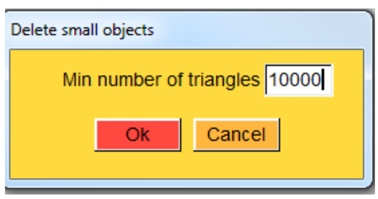


Figure 8.22: Delete small objects window

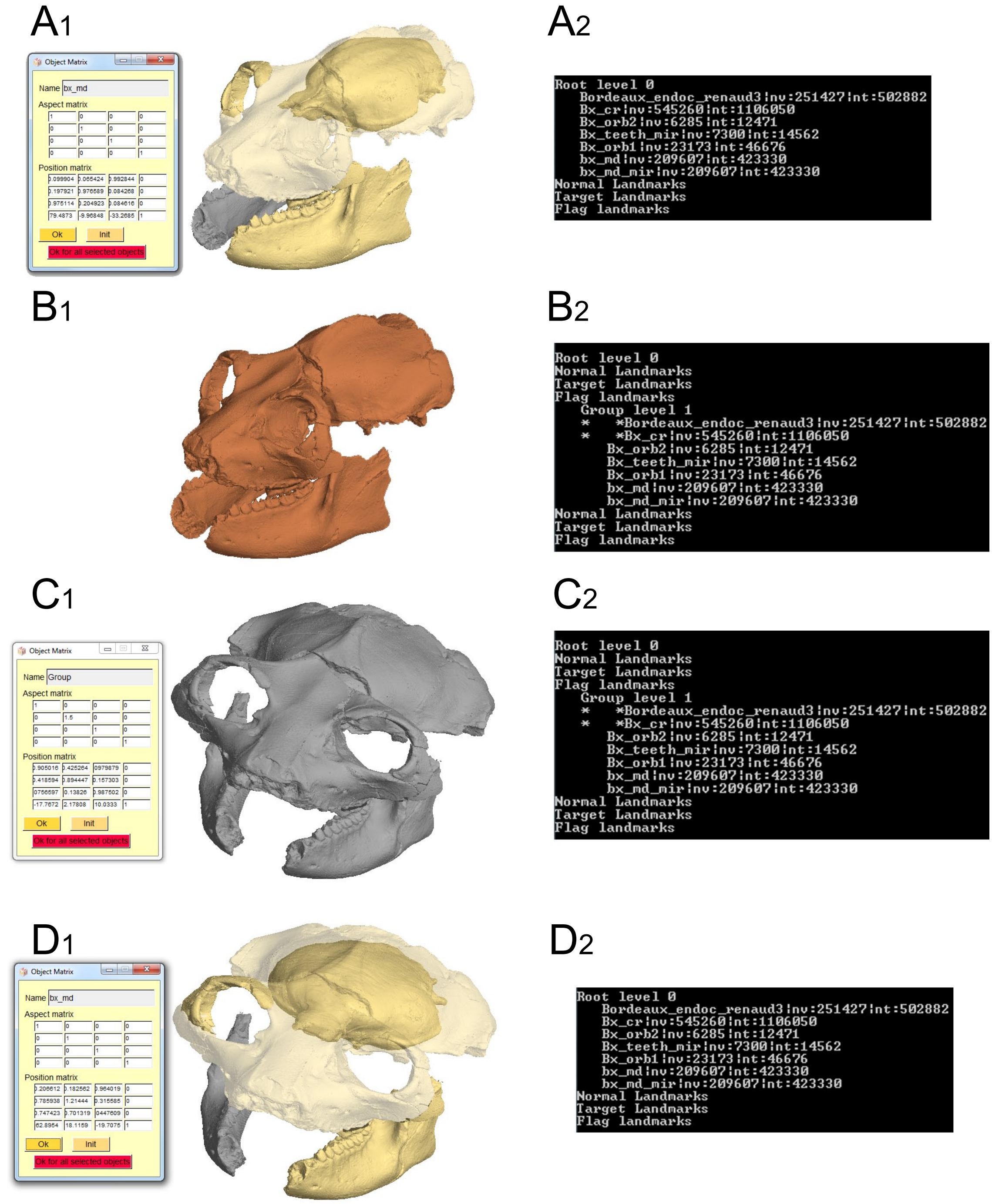


Figure 8.21: Grouping objects. A1: 7 objects are opened. Original matrices of "bx_md" object are shown (right mandible, selected). A2: corresponding display object order (Show →Display object order). B1: The 7 object are grouped. The group is drawn in brown. B2: corresponding display object order. C1: the position of the group is changed, as well as its aspect matrix. Note that the enclosed meshes cannot be saved in that state. C2: corresponding display object order. D1: objects are ungrouped. Modified matrices of "bx_md" object are shown (right mandible is selected). Now, the 7 meshes can be saved. D2: corresponding display object order (Show →Display object order).

8.5.2 Threshold : a given volume

Using this option, you may delete selected objects smaller than a given volume.

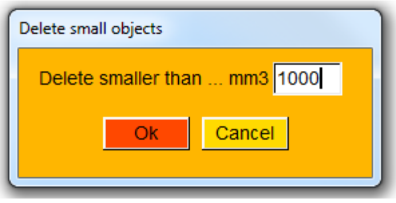


Figure 8.23: Delete small objects window 2

Chapter 9

Landmarks

Contents

As stated above, landmarks can be set on surfaces by pressing “L" + left mouse click. Several actions can be performed on landmarks.

9.1 Select a given range of landmarks.

By opening the “select landmark range" window, you can select a given range of landmarks. This option may be useful when you need to save only a a specific sub-range of all digitized landmarks.

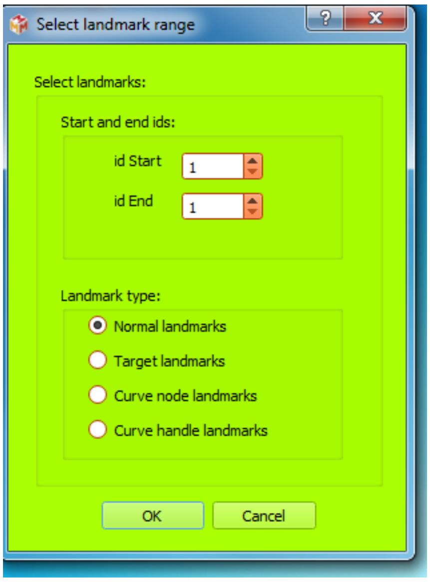


Figure 9.1: Select landmark range window

9.2 Push back selected landmarks on closest surface.

When set via pressing “L" + left click, landmarks are positioned on one surface’s vertex. Selected landmarks can be subsequently moved manually to other locations (for instance, if you want to place a given landmark in the middle of a canal or a foramen, or between two unfused bones). However, you may sometimes want to push back automatically some selected landmarks to the position of the closest surface’s vertex available. This can be achieved using this option.

9.3 Edit all selected flag landmarks.

Using this option, you can modify the length and the color of several selected flag landmarks at once.

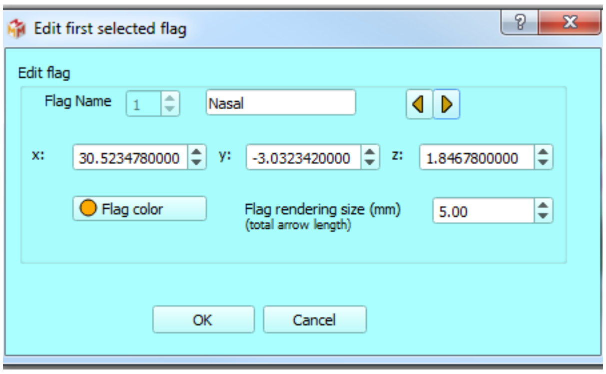


Figure 9.2: Edit all selected flags window

9.4 Change selected landmarks orientation according to surface normals.

When set via pressing “L" + left surfaces, landmark orientation is that of the vertex on which it is placed. Selected landmarks’ orientation can be subsequently moved manually. However, you may sometimes want to reset one or several landmarks’ orientation automatically to that of the closest surface’s vertex available. This can be achieved using this option.

9.5 Landmarks involved into curves.

9.5.1 Move curve handles (selected yellow landmarks) semi-automatically

This option allows saving a lot of time when creating 3D Bezier curves with ISE-MeshTools (see Fig. 9.4). Also see “working with curves" tutorial for further details regarding curve digitization with ISE-MeshTools).

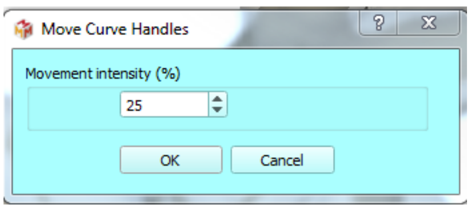


Figure 9.3: Mode handles window

Requirement : at least a handle landmark (“target" landmark) must be selected. Depending on whether selected curve handles lie within the curve, at the start of the curve or at the end of the curve, their displacements differ (see Fig. 9.5)

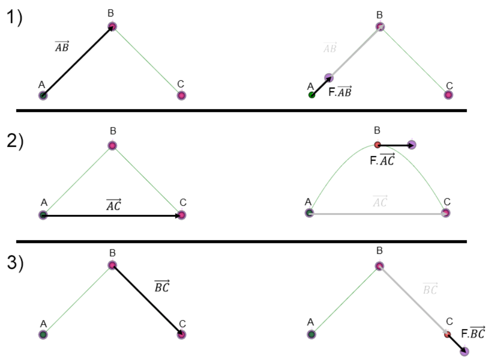


Figure 9.4: Moving handles semi-automatically.
1. Left: curve handle is associated to a curve starting point (A), and a following point (B) exists. Vector $\overrightarrow{AB}$ is computed, as well as its length |AB|. Right: curve handle associated to A is moved from point A along $\overrightarrow{AB}$. Displacement length=movement intensity/|AB|.
2. Left: curve handle is associated to a point B lying between two points (A and C). Vector $\overrightarrow{AC}$ is computed, as well as its length |AC|. Right: curve handle associated to B is moved from point B along $\overrightarrow{AC}$. Right: displacement length=movement intensity/|AC|.
3. Left: curve handle is associated to a curve ending point (C), and a preceding point (B) exists. Vector $\overrightarrow{BC}$ is computed, as well as its length |BC|. Right: curve handle associated to C is moved from point C along $\overrightarrow{BC}$. Displacement length=movement intensity/|BC|.

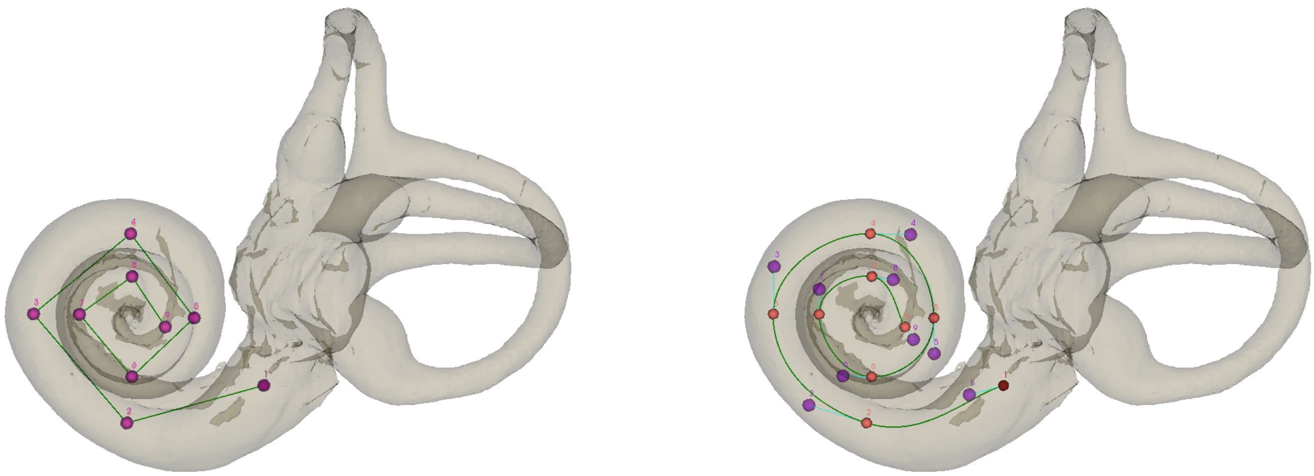


Figure 9.5: Example of curve handles semi-automatic displacement (movement intensity: 25%).

9.5.2 Normal landmarks (red): define as curve starting points (green)

Selected landmark will be given flag “1"

9.5.3 Normal landmarks (red): connect to preceding starting points (violet)

Selected landmark will be given flag “3".

9.5.4 Normal landmarks (red): define as curve milestones (blue)

Selected landmark will be given flag “2".

9.5.5 Green, blue, violet landmarks: set back to normal landmarks (red)

Selected landmark will be given flag “0".
Further information regarding curve use in ISE-MeshTools is available in the section “File →
Further information regarding curve use in ISE-MeshTools is available in the section “File →
Curves" section (section 6.6) and in the tutorial “working with curves".

Chapter 10

Scalars

Contents

Scalar values can be associated to each vertex. Scalar values can be displayed when scalar display mode is active. To activate/deactivate scalar display mode, click on "". When active, the rainbow color scale (see Fig. 10.1) shows up in the bottom-right part of the 3D rendering window.

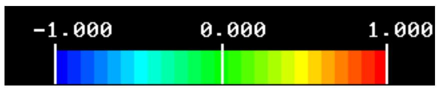


Figure 10.1: Rainbow color scale, showing a "min" range display value of -1, a "max" range display value of 1, and a middle range display value of 0.

10.1 Show scalar rendering options window.

Displayed scalars and scalar associated color scales can aslo be onpened by clicking on "" (see Fig. 10.2).

__**Available controls:**__

Chose active scalar: please chose among the available scalars (see below for further information).

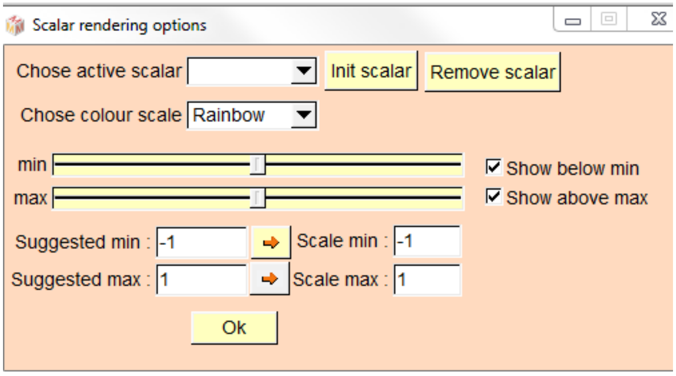


Figure 10.2: Scalar rendering options window

Chose color scale: please chose among the available color scales (see below for further information).

Init scalar: set current active scalar values to “0" for all selected objects. Remove scalar : removes active scalar for all selected objects. This option is useful if you plan to save surfaces in the .vtk format and do not want ISE-MeshTools to save associated scalar values (this will save some disk space).
min: changes the minimal value of the active color scale.
max: changes the maximal value of the active color scale.
Show below min: if active, all vertices with an associated active scalar value below “min" will be drawn using the color situated at the leftmost part of the active color scale. If not, these vertices will be transparent.
Show below max: if active, all vertices with an associated active scalar value above “max" will be drawn using the color situated at the rightmost part of the active color scale.
Suggested min: suggested “min" range display value. This value is computed in order to use the color scale at its best.
Suggested max: suggested “max" range display value. This value is computed in order to use the color scale at its best.
“ ": Set min/max to suggested min/max, respectively.
Scale min: current “min" range display value.
Scale max: current “max" range display value. *Ok*: applies changes.

10.2 Scalars: distance from camera (Depth).

Computes distance from camera for all vertices of all selected objects. This option may offer a better perception of the 3D structure of an object on a 2D screen representation.

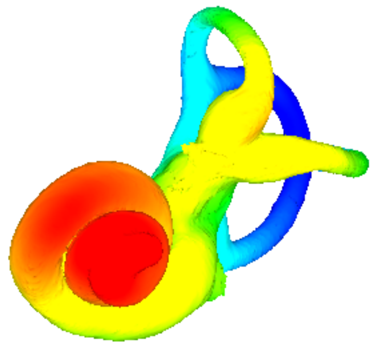


Figure 10.3: Example of 3D rendering of “Depth" scalars. Scalar mode is active, the rainbow color scale is used.

10.3 Scalars : compute vertice curvature.

vtkCurvatures filter is implied in this option.
vtkCurvatures filter offers 4 ways to compute surface’s curvature at each vertex (see Fig. 10.5):
- Principal maximal curvature
- Principal minimal curvature
- Gaussian curvature
- Mean curvature.
See vtkCurvatures’ documentation for further details.

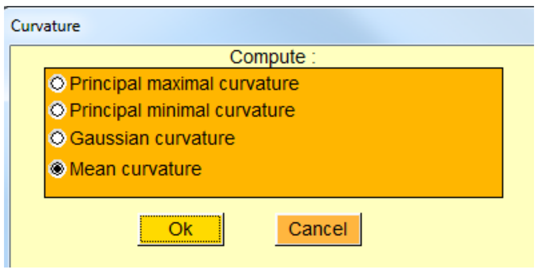


Figure 10.4: Curvature window

10.4 Scalars: compute thickness.

Thickness within an object (see Fig. 10.8) is defined the following way: for a given vertex, the minimal distance between this vertex and other vertices in the direction opposite to that of the surface’s normal is computed. In order to minimize computation time, a maximal distance (Maximal thickness (mm)) is asked to the user, in order to reduce the amount of vertices investigated at a given location.

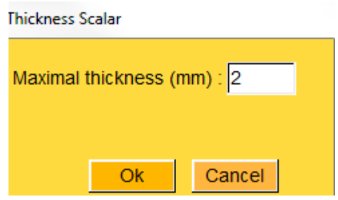


Figure 10.6: Thickness scalar window

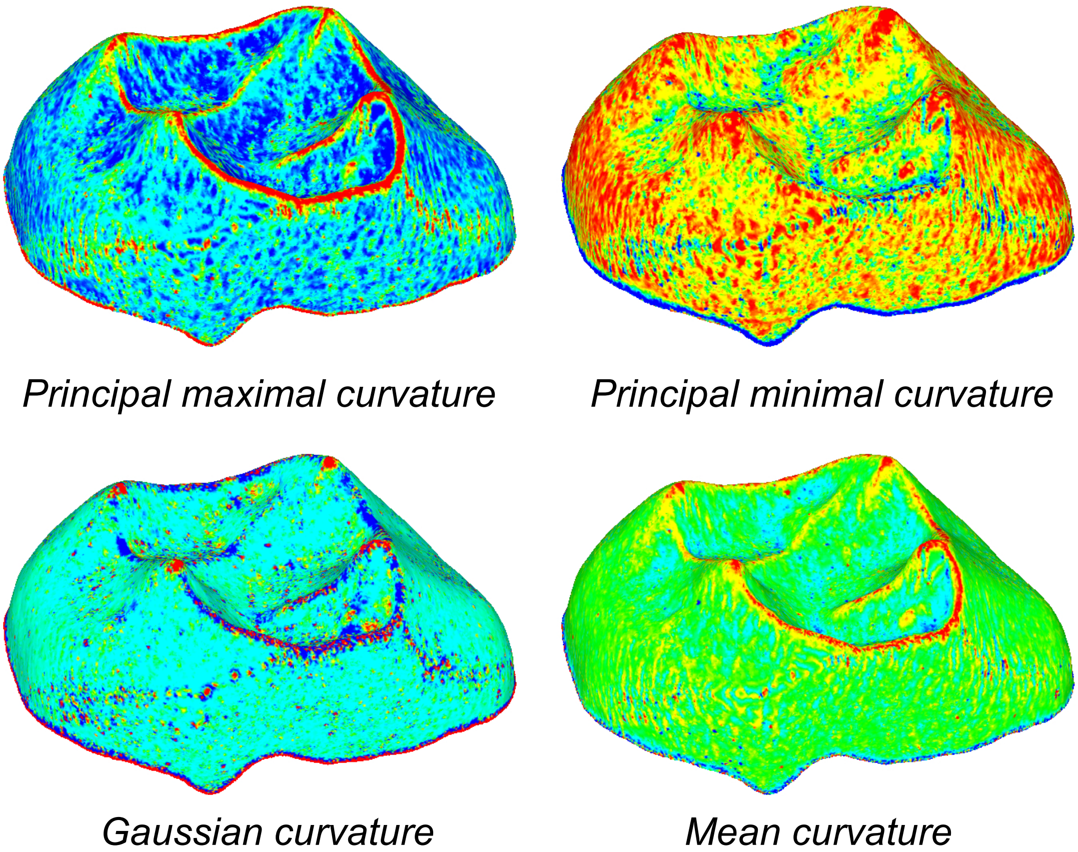


Figure 10.5: Examples of 3D rendering of “Curvature" scalars. Scalar mode is active, the rainbow color scale is used. Specimen : enamel dentine junction (EDJ) of the second superior molar of a juvenile medieval human from Sains-en-Gohelle (France). Specimen number : SP07. Image credit: Mona Le Luyer (PACEA, Bordeaux).

10.5 Scalars: compute thickness between two objects.

Thickness between tow objects is defined the following way (see Fig. 10.9): for a given vertex of the impacted object, the minimal distance between this vertex and other vertices of the observed surface in the direction opposite to that of the impacted surface's normal is computed. Again, in order to minimize computation time, a maximal distance (Maximal thickness (mm)) is asked to the user, in order to reduce the amount of vertices investigated at a given location. Only selected surfaces appear in the impacted object and observed object lists.

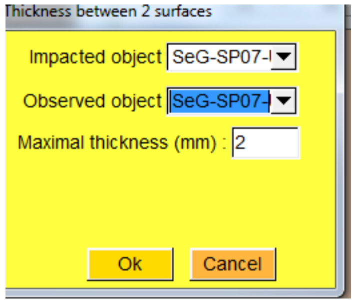


Figure 10.7: Thickness between 2 surfaces window

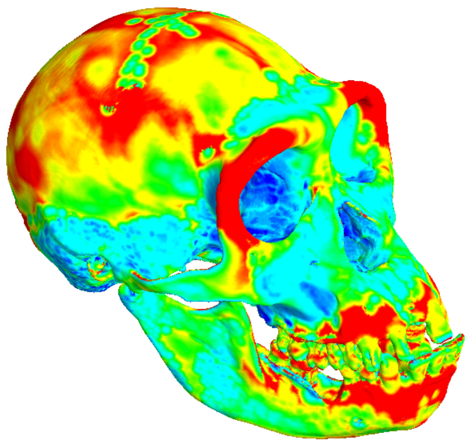


Figure 10.8: Example of 3D rendering of "Thickness" of the 3D model ot the type specimen of *Pan paniscus* (downloadable on Metafro.be). Scalar mode is active, the rainbow color scale is used. Thickness between 2 surfaces window

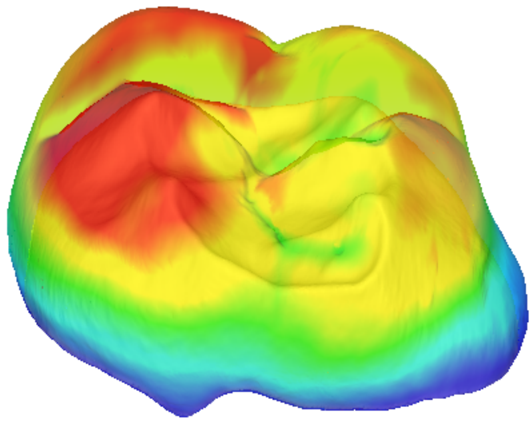


Figure 10.9: Example of 3D rendering of "Thickness" between two objects' scalars. Scalar mode is active, the rainbow color scale is used. Impacted object : enamel surface of SP07 specimen (see above for details). Observed object : enameldentine junction's surface (EDJ) of SP07 specimen.

10.6 Smooth active scalars (Gaussian blur).

Active scalars are "smoothed" the following way : for each vertex, a new scalar value is computed as the mean of the scalar values of all neighbouring vertices (see Fig. 10.10).

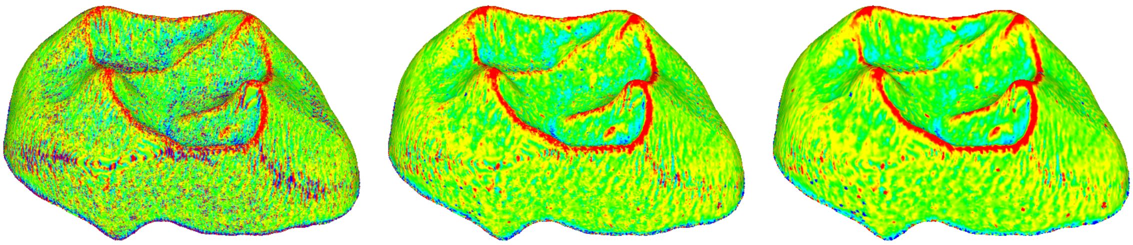


Figure 10.10: Smoothing scalars. Examples of 3D rendering of “Mean Curvature" scalars. Scalar mode is active, the rainbow color scale is used. Left : “raw" mean curvature. Middle : mean curvature scalars smoothed once. Right : mean curvature scalars smoothed twice. Specimen: EDJ of SP07 specimen.

10.7 Init RGB.

Whenever colors are associated to vertices in .PLY of .VTK files (this can be achieved for instance by painting on meshes with Meshlab or by transferring textures obtained by laser scanning to the vertices), ISE-MeshTools stores this “initial RGB" color information inside a dedicated scalar (called “Init_RGB"). See for instance Fig. 10.11.
Note: you have to select a Mesh in order to activate this option.

10.8 Saving and loading scalars.

Computed scalars can be saved inside the .vtk surface files. In order to access scalar values into other software (such as a text editor), save the .vtk files in ASCII format. Saved scalars can be reloaded into ISE-MeshTools. Saving surfaces into .vtk format provides an efficient means to store and exchange computed scalars.

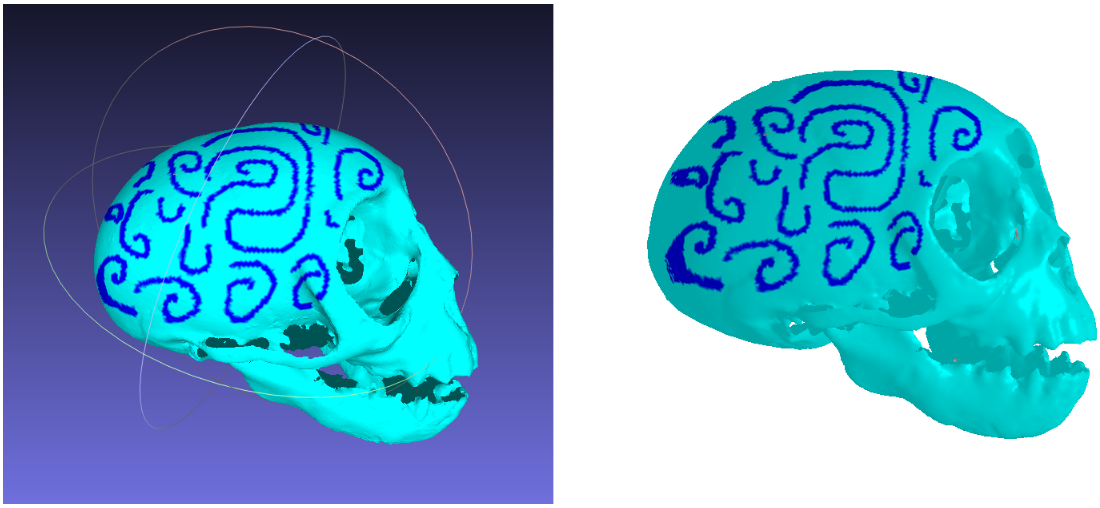


Figure 10.11: Initial RGB Scalar. Left: a mesh is painted using Meshlab. Right: the same mesh opened with ISE-MeshTools using the “Init RGB" option.

Chapter 11

Tags

Contents

11.1 Tagging surfaces with ISE-MeshTools.

25 tags (ordered between 0 and 24) can be given a label, a color, a transparency, and be manually delimitated. A greater number of tags can be given to a surface using automatic tagging tools such as the “Tag connected regions" tool, but these additionnal tags can not be labeled or manually edited. As stated earlier, in order to edit tags, it is advisable to activate the “tag mode" (press “" button). In this mode, selected surfaces (on which you can interact and tag) can be drawn according to tag values at each vertex if “tag display mode"is active. Press “" to activate “tag diplay mode". When active, the tag color scale shows up in the bottom-right part of the 3D rendering window (see Fig. 11.1). Selected surfaces can be tagged using the pencil tag tool (“"), the magic wand tag tool (“"), the paint bucket tag tool (“") or the lasso tag tool (“").

Figure 11.1: Example of tag color scale

11.1.1 Pencil tag tool

Pencil tag tool controls:
T pressed + left mouse click : tags the selected surface using currently active tag.
T pressed + right click : tags the selected surface using tag 0 (usually used as “exterior" tag). This option is often used to “clear" a wrongly tagged part.
Pencil tag special option:
pencil tag size can be modified in the Tag option window. This option defines the tag propagation extension level, which starts at the vertex on which the mouse click is performed.

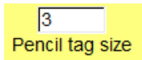


Figure 11.2: Pencil tag size option

11.1.2 Magic wand tag tool

Magic wand tag tool controls:
T pressed + left mouse click : tags the selected surface using currently active tag. T pressed + right click : tags the selected surface using tag 0 (usually used as “exterior" tag). This option often is used to “clear" a wrongly tagged part.

Magic wand special option:

Magic wand limit angle size can be modified in the Tag option window.

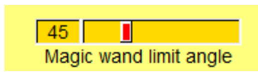


Figure 11.3: Magic wand limit angle

This option defines the tag propagation extension level, which starts at the vertex on which the mouse click is performed : propagation stops if the angle between the notmal of current vertex and the normal of the vertex on which the mouse click was done is found is greater than the defined angle.

11.1.3 Paint bucket tag tool

Paint bucket tag tool controls:

T pressed + left mouse click : tags the selected surface using currently active tag.
T pressed + right click : tags the selected surface using tag 0 (usually used as “exterior" tag). This option is often used to “clear" a wrongly tagged part.
Note : the paint bucket tag tool works exactly like the magic wand tag tool when the magic wand limit angle is set to 180°.

11.1.4 Pencil tag, magic wand and paint bucket common option

Magic wand special option:

These 3 tools share a common option, available in the Tag option window, the “allow color override" checkbox.

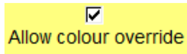


Figure 11.4: Allow color override option

- If checked : no attention is paid to the color of the vertex on which the left/right click was done during tag propagation.
- If unchecked: tag propagation will stop if a tag color different from 0 (exterior tag) and from that of the vertex on which the mouse click was done is found.
Important note: be careful when using “allow color override" option checked with the paint bucket tag tool or with the magic wand, as it will paint a large region uniformly (minutes or even hours of work may be lost in a single click, if you did not save your tagged surface in .vtk format earlier).

11.1.5 Lasso tag tool

The lasso tag tool can be used alone (option 1) or in combination with the pencil, the magic wand or the paint bucket (option 2). Once “lasso tag tool" button is pressed, the mouse cursor changes to a cross in the 3D window. Additional mouse and keyboard controls become available (see Table 11.1). Once option 1 or option 2 is achieved, the usual mouse and keyboard controls become available again.

11.2 Show tag options window.

The “Tag options" window (see Fig. 11.5) can also be opened by clicking on “" . **Available controls:**

Define tag labels, colors and transparency group:
Labels: you may define tag labels for all 25 available tags.
Active tag: you may define the currently active tag.
color: you may define the color for all 25 available tags.

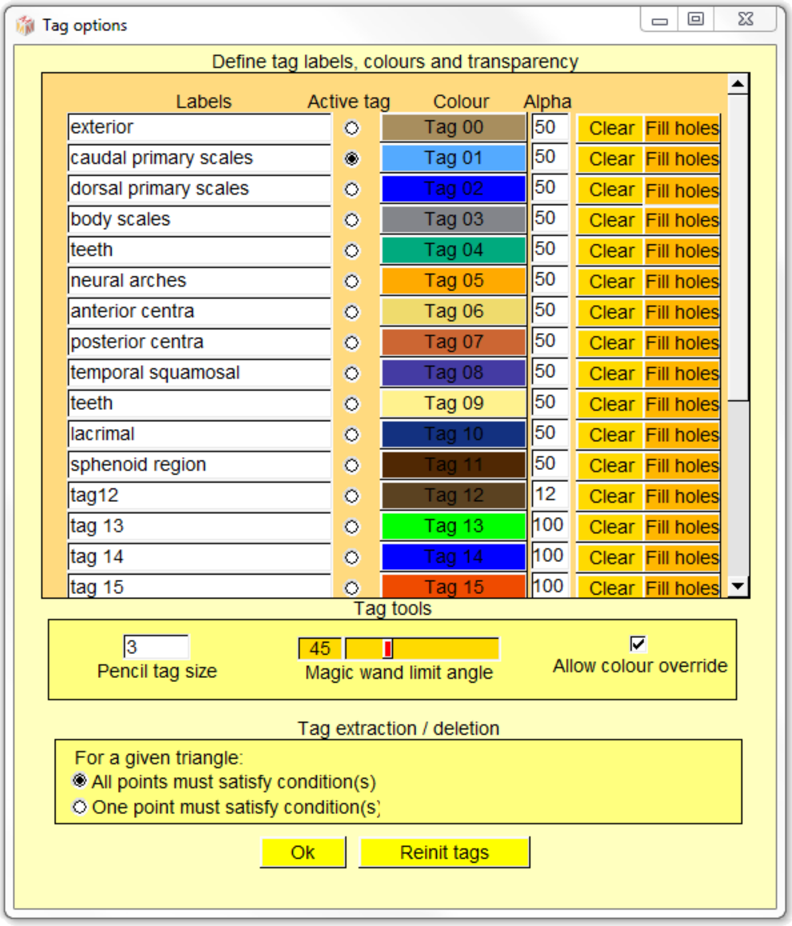


Figure 11.5: Tag options window.

Left click	Adds a segment to polygon (segments are drawn yellow)
Right click	Connects last segment to first segment. If two segments cross each other, lasso tag action is cancelled. Otherwise, the closed polygon is drawn red.
Option 1: once polygon is closed, middle click or "C" + right click.	when the click falls **inside**/*outside* the closed red polygon: the region falling inside the polygon **is tagged**/*is not tagged* using the active tag color, respectively. The region falling outside the polygon **is not tagged**/*is tagged* using the active tag color, respectively.
Option 2: once polygon is closed: Press "T" + left or Press "T" +right click	The selected tag tool (magic wand or paint bucket) is be used, but tag propagation will not cross over polygon edges.

Table 11.1: Lasso tag controls

Alpha: you may define the transparency for all 25 available tags.
Clear: clears the tag region (all vertices of this region will be set to 0 = Tag 00). *Fill holes:* opens the "Fill holes" window (see Fig. 11.6). Pressing ok will fill all regions adjacent to the concerned tag region (Tag id) containing less than "Max num" vertices (see for instance Fig. 11.7).
Tag tools group: Pencil tag size: This option defines the tag propagation extension level of the pencil tag tool. Magic wand limit angle : This option defines the tag propagation extension level of the magic wand tag tool (see above for further explanations).
Allow color override: The pencil, magic wand and paint bucket tag tools share this option. If active, tag propagation will stop if a tag color different from 0 (exterior tag) and from that of

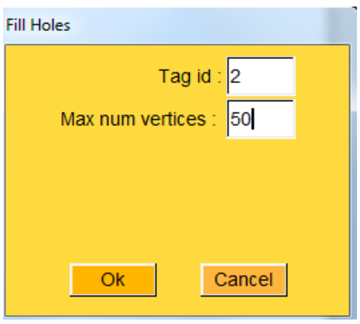
Holes.png


Figure 11.6: Fill Holes window.

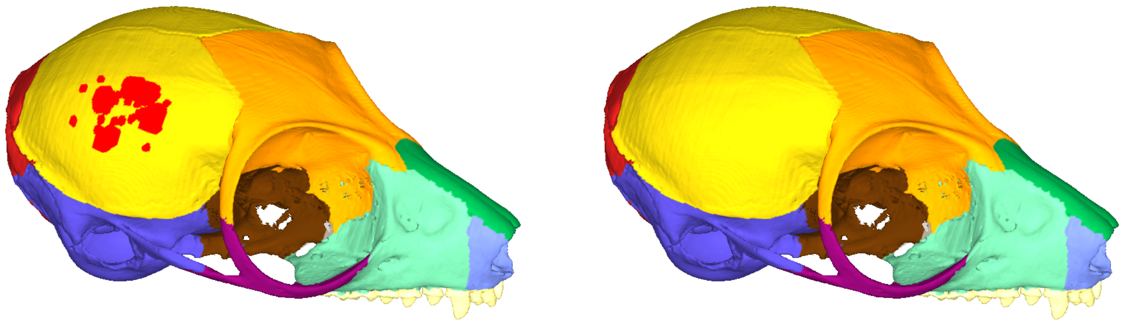


Figure 11.7: Example of tag hole filling. Left : cranium of *Microcebus murinus* presenting a parietal region tagged mostly in yellow, and presenting red “holes". Right : “holes" present in the yellow region were filled using “Max num vertices" = 4000 option.

the vertex on which the mouse click was done is found.

Tag extraction/deletion:
For a given triangle : *All points must satisfy condition* / **One point must satisfy condition.** This option defines the extent to which region extraction or deletion is performed at the boundaries of the concerned tagged region.

Example of tag merging. Left : cranium of *Microcebus murinus* presenting the parietal region tagged in yellow, the frontal region tagged in orange. Right : frontal tag region was merged into the parietal region.

11.3 Convert RGB colors to Tags.

This option (see Fig. 11.8) may be useful when you just have opened a .ply file already containing RGB colors (for instance, let us suppose that you have painted a surface using MeshLab software, and that you wish to convert those colors into tags). RGB colors contained in .ply files are inserted inside the RGB scalar when opened with MeshTools. I you plan to transform those RGB values into tags with MeshTools, be aware of the fact that RGB scalar is reinitialised extremely frequently: for instance as soon as you activate the tag or the scalar display mode, or whenever you change the objects' color rendering. So I advise you to use the present option only immediately after opening the surface. You have two options :
1) exact color match
→ In that case, in order to be given a tag value other than Tag 00, a vertex must satisfy the following condition: its RGB scalar value should match one of the 25 colors defined in the “Tags options" window. If a vertex does not satisfy this condition, it is given the Tag 00 value.

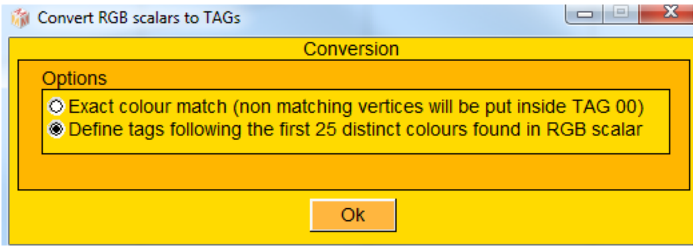


Figure 11.8: Convert RGB scalars to TAGS window

2) Define tags following the first 25 distinct colors found in RGB scalar

→ In that case the following sequence of operations is applied:
a) ISE-MeshTools searches for the first 25 distinct RGB colors inside the object; ISE-MeshTools gives them numbers ranging from 0 to 24. Important : if more than 25 distinct colors exist in the RGB scalar, they are not given a tag value.
b) ISE-MeshTools updates tag colors according to the 25 first colors found in the RGB scalar.
c) All vertices are given a Tag value following the procedure defined above in #1 (exact color match).
Advantages of #1 : if you have well prepared your RGB colors so that they matches well those associated with the 25 tags, this procedure will work perfectly.
Advantages of #2 : if you have up to 25 distinct RGB colors in your .ply file and if you do not want to bother to edit manually the 25 tags and give them a RGB color, you will save time with this procedure.
Drawbacks of both methods: if your file contains more than 25 distinct RGB colors, you will definitely lose information in the process.
Be also aware that when saving .ply files with ISE-MeshTools, the RGB coulours saved within the file are those currently rendered in the 3D screen. If you spent time to color a 3D surface in .ply format (for instance with MeshLab) and you have opened it subsequently with ISE-MeshTools and have saved it again, there is a high probability that all the initial RGB colors will have been replaced with those rendered in ISE-MeshTools.

11.4 Merge tags.

Two tagged regions can be merged into a single one All source tags will be put into target tags.

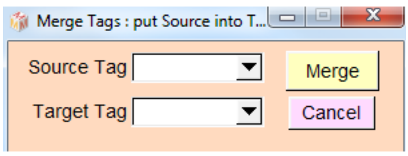


Figure 11.9: Merge tags window

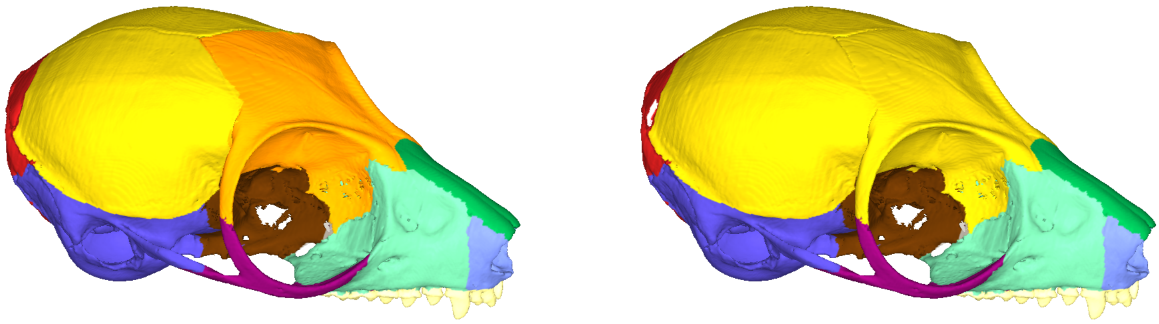


Figure 11.10: Example of tag merging. Left : cranium of *Microcebus murinus* presenting the parietal region tagged in yellow, the frontal region tagged in orange. Right : frontal tag region was merged into the parietal region.

11.5 Tag connected regions.

This option implies vtkPolyDataConnectivityFilter. This filter will tag all non-connected regions of the selected input surface into different colors (see Fig. 11.11).

11.6 Extract.

Note: The "Tag extraction/deletion" option in the Tag options window will affect the boundaries of the extracted regions (see Fig. 11.12).

11.6.1 Extract active tag corresponding region

Using this window, a single mesh will be created out of the active tag corresponding region of the selected object.

11.6.2 Extract all tagged regions as several new objects

Using this window (see 11.13), all meshes will be created out all different existing tags. In order to prevent extremely small surface objects to be created, a minimal region size parameter for tag extraction is asked. For an example, see Fig. 11.14.

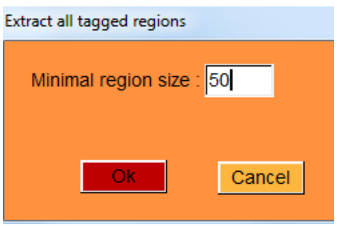


Figure 11.13: Extract all tagged regions window.

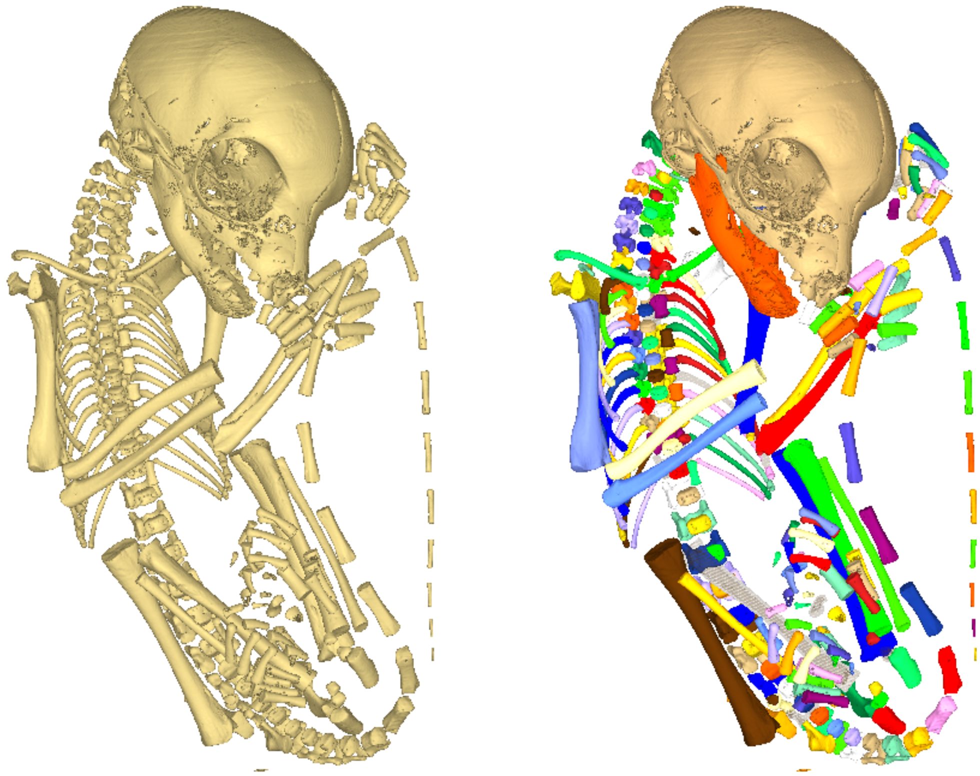


Figure 11.11: Left: original surface. Right: the same mesh automatically tagged into 304 non connected regions.

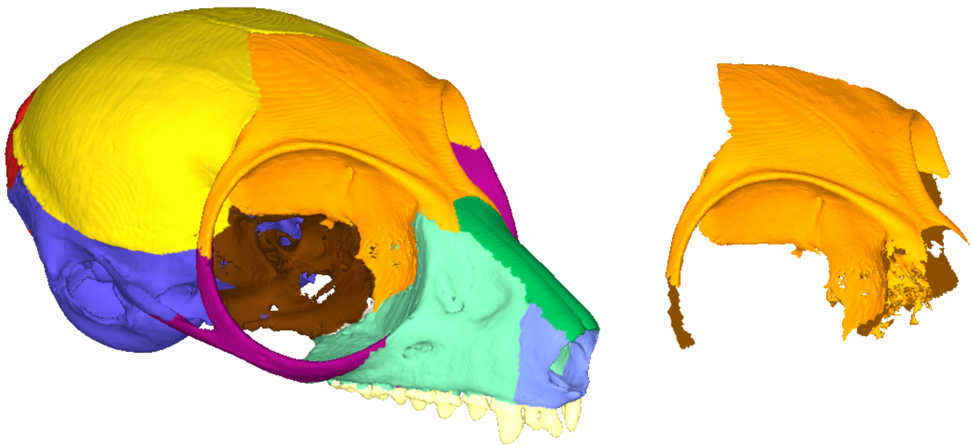


Figure 11.12: Example of tag extraction. Left : cranium of *Microcebus murinus* presenting the frontal region tagged in orange. Right : frontal tag region was extracted into a new surface object.

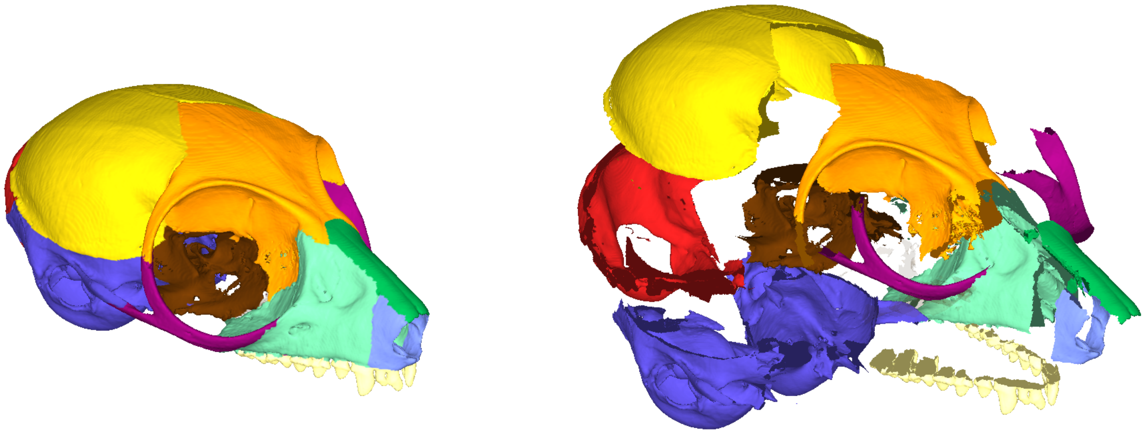


Figure 11.14: All tags extraction. Left: original tagged surface. Right: all tagged regions were extracted into single independent surface objects

11.6.3 Extract one tagged region as 1 new object

This option works similarly as the “Extract active tag corresponding region" option mentioned above, except that you can reach tag id values beyond the 25 reachable in the Tag options window.

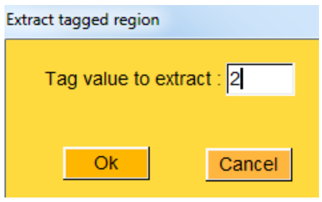


Figure 11.15: Extract tagged region window.

11.6.4 Extract tag or other scalar range as 1 new object

Using this option, you may extract several tagged regions into one single mesh, or all regions ranging from a minimal value up to a maximal value for a given scalar into a single new mesh. See for instance Fig. 11.17.

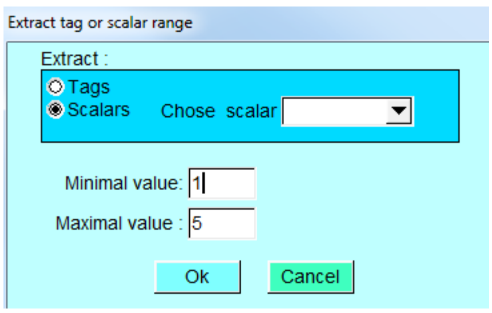


Figure 11.16: Extract scalar range window.

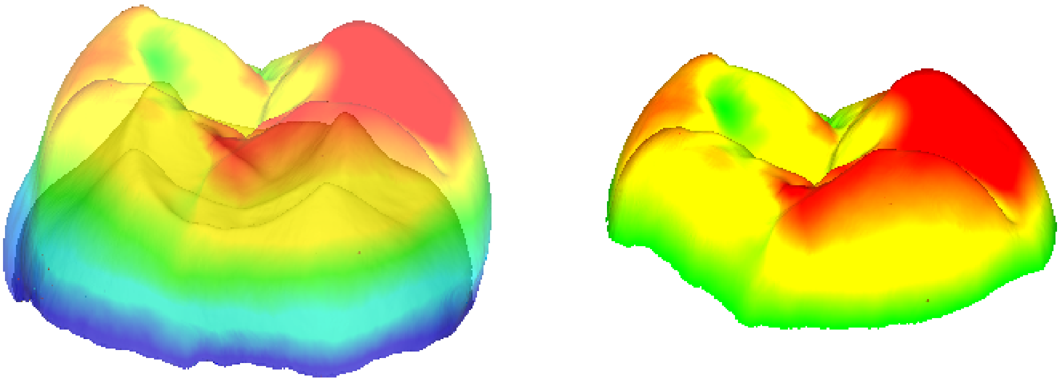


Figure 11.17: Extraction of the region showing an enamel thickness greater than 1 mm. Specimen : enamel surface of SP07 specimen (see “Scalars" and “Acknowledgement" sections for details regarding this specimen). Image credit : Mona Le Luyer (PACEA, Bordeaux).

11.7 Delete.

Note: The “Tag extraction/deletion" option in the Tag options window will affect the boundaries of the deleted regions.

11.7.1 Delete one tagged region

This option works similarly as the “Extract one tagged region" option mentioned above, except that it deletes the selected tag region.

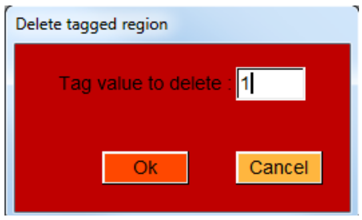


Figure 11.18: Delete tagged region window.

11.7.2 Delete active tag corresponding region

This option works similarly as the “Extract active tag corresponding region" option mentioned above, except that it deletes the corresponding active tag region.

11.7.3 Delete all tagged regions except TAG 00

This option deletes all tagged region except the vertices tagged with TAG 00 (exterior).

Chapter 12

Show

Contents

12.1 General options.

The general options window (see Fig. 12.1) contains the following sections:

12.1.1 Behavior when opening surface

Object position:
- *Move object center of mass at x=0, y=0, z=0*: when active, the position matrix of a newly opened surface is set in order to display the object at the origin of the coordinate system (x=0, y=0, z=0). This option is useful when surface native coordinate system is far from the origin (this is often the case if you see nothing in the 3D rendering window after opening a surface).
- *Keep object native coordinate system*: when active, the position matrix of a newly opened surface is set to the identity matrix.

Camera:

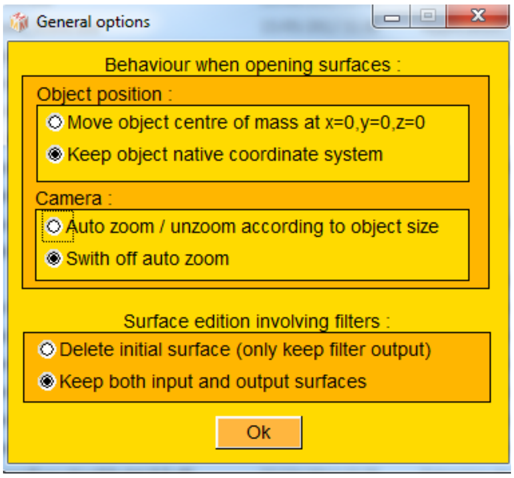


Figure 12.1: General options window.

- *Auto zoom / unzoom according to object size*: when active, the zoom of the camera is modified in order to match the object global size.
- Switch off auto zoom: switches off the preceding option.

12.1.2 Surface edition involving filters

The following "filters" are available in ISE-MeshTools (see "Edit selected surfaces" section):
- Connectivity filters
- Lasso cut
- TPS deformation
- Mesh mirroring/smoothing/decimation/densification/hole filling

Delete initial surface (only keep filter output): when active, when using one of the previously mentioned filters, the initial object is deleted. Only the filter output is kept. This option is useful to avoid object multiplication.

Keep both input and output surfaces: when active, when using one of the previously mentioned filters, the initial object is kept. This option is useful to compare the initial object and the filter output.
Note that filter output object's name differs from filter input objects' name.

12.2 Show object view/hide window.

This window (see Fig. 12.2) shows the list of currently opened meshes. You may hide / show objects using this interface. Note that if you delete/load objects while opened, the list will not

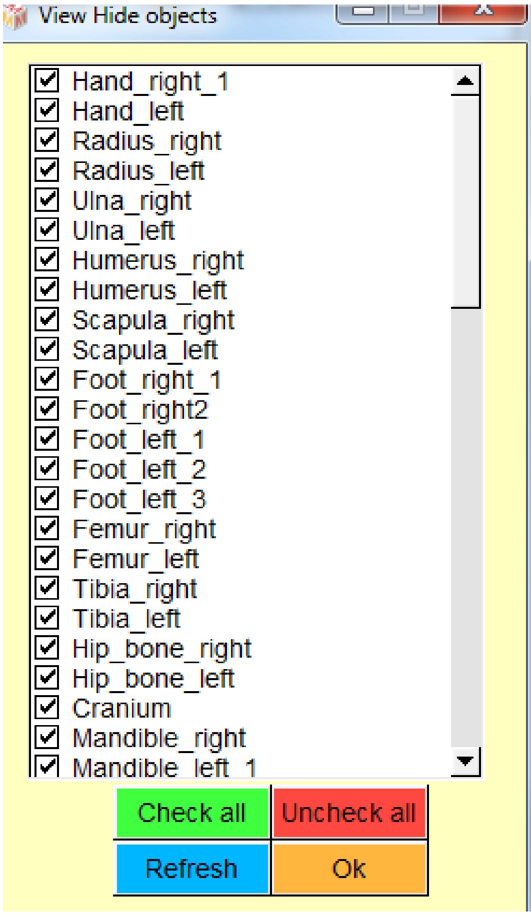


Figure 12.2: View Hide Objects window.

be updated. Please click on “refresh" to see the actual current list of existing objects.

“Check all" : will show all objects

“uncheck all" : will hide all currently loaded objects.

This window is useful to hide/show structures.

12.3 Area and volume of selected objects.

This option shows in the output window the list of surface objects which are selected.

12.4 Print scalar list.

This option shows in the output window the list of existing scalars. When saving .vtk files, all these scalars are saved inside the .vtk file along with the geometry of the object(s). When saving .ply files, only the RGB scalars are saved.

12.5 Print list of selected objects.

This option shows in the output window the full list of the names of all currently selected surfaces.

12.6 Print list of distinct selected objects.

This option shows in the output window the list of the distinct names of all currently selected surfaces. It may sometimes be useful to compare the full list of names and the list of distinct names of selected objects when working with projects : ISE-MeshTools only allows to save series of surface files having distinct names (see tutorial “Working with projects" for further explanations).

12.7 Show object display order.

This option shows in the output window the list of objects (landmarks, target landmarks, flag landmarks, logical objects, surfaces) loaded into ISE-MeshTools, as well as their display order.

Chapter 13

About(?)

Contents

13.1 About

This window (see Fig. 13.1) shows the current version of ISEMeshTools.

Figure 13.1: About window.

Chapter 14

Acknowledgments

Contents

14.1 Design concepts

I wish to thank Christoph Zollikofer and Marcia Ponce de León, from whom I borrowed many of the design concepts implemented in FoRM-IT (Fossil Reconstruction and Morphometry Interactive Toolkit; [Zollikofer and Ponce de León, 1995, 2005]).

14.2 Code development

I wish to thanks Cécile Peladan, who has successfully implemented GPU acceleration and ICP registration functions, and Gérard Subsol, with whom I supervised Cécile during her Master Thesis project. Many thanks to Stefan Schlager and Jean Dumoncel for their helpful comments and their contribution to the cross-platform compatibility of the source code.

14.3 Specimens illustrated

I wish to thank the curators of different institutions for the access to most specimens illustrated in this document: Nathalie Mémoire from the Musée d'Histoire Naturelle de Bordeaux, Christoph Zollikofer and Marcia Ponce de León from the Anthropological Institute and Museum of Zürich, Jacques Cuisin from the Musée d'Histoire Naturelle de Paris, Peter Giere and Robert Asher from the Museum für Naturkunde (Berlin), Suzane Jiquel from the ISEM, Nadine Mestre-Francès from the Université Montpellier II. Thanks also to Mona Le Luyer and Pauline Colombet (PACEA, Bordeaux) , the PEPS IdEx Bordeaux / CNRS 3 Dent'in for the access to the digital reconstruction of the specimen SP07 of Sains-en-Gohelle, and Cédric Beauval and Archéosphère SARL (Bordeaux) for the access to these remains.

14.4 3D data acquisition facilities

I wish to thank the MRI Platform (Montpellier), Zollikofer and Marcia Ponce de León from the Anthropological Institute and Museum (Zürich), Paul Tafforeau and the E.S.R.F. (Grenoble, France) for the access to the scanning facilities in which the illustrated specimens were digitized.

References

Lisa S Avila, Sébastien Barré, Rustuy Blue, Diavid Cole, Berk Geveci, William A Hoffman, Brad King, C Chalres Law, Kenneth M Martin, William J Schroeder, and Amy H Squillacote. *The VTK User's Guide*. 2001. ISBN 1930934238. URL www.vtk.org.

Christoph P E Zollikofer and Marcia S. Ponce de León. Tools for rapid prototyping in the biosciences. *IEEE Computer Graphics and Applications*, 15(6):48–55, 1995. ISSN 02721716. doi: 10.1109/38.469515.

Christoph P E Zollikofer and Marcia S Ponce de León. *Virtual Reconstruction: A Primer in Computer-Assisted Paleontology and Biomedicine*. Wiley, New York, 2005. ISBN 978-0-471-20507-4.